Gordan Jezic Mario Kusek
Ngoc Thanh Nguyen Robert J. Howlett
Lakhmi C. Jain (Eds.)

LNAI 7327

Agent and Multi-Agent Systems
Technologies and Applications

6th KES International Conference, KES-AMSTA 2012
Dubrovnik, Croatia, June 2012
Proceedings

Springer

Lecture Notes in Artificial Intelligence 7327

Subseries of Lecture Notes in Computer Science

Gordan Jezic   Mario Kusek
Ngoc Thanh Nguyen   Robert J. Howlett
Lakhmi C. Jain (Eds.)

Agent and Multi-Agent Systems

Technologies and Applications

6th KES International Conference, KES-AMSTA 2012
Dubrovnik, Croatia, June 25-27, 2012
Proceedings

Springer

Volume Editors

Gordan Jezic
Mario Kusek
University of Zagreb, 10000 Zagreb, Croatia
E-mail: {gordan.jezic, mario.kusek}@fer.hr

Ngoc Thanh Nguyen
Wroclaw University of Technology, 50-370 Wroclaw, Poland
E-mail: ngoc-thanh.nguyen@pwr.edu.pl

Robert J. Howlett
KES International, Shoreham-by-sea, BN43 9AF, UK
E-mail: rjhowlett@kesinternational.org

Lakhmi C. Jain
University of South Australia, Adelaide, SA 5095, Australia
E-mail: lakhmi.jain@unisa.edu.au

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-30946-5 e-ISBN 978-3-642-30947-2
DOI 10.1007/978-3-642-30947-2
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012939053

CR Subject Classification (1998): I.2.11, I.2, H.3.4-5, H.4, D.2, J.4, I.6

LNCS Sublibrary: SL 7 – Artificial Intelligence

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

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

Preface

This volume contains the proceedings of the 6th KES Conference on Agent and Multi-Agent Systems – Technologies and Applications (KES-AMSTA 2012) held at the Centre for Advanced Academic Studies (CAAS) in Dubrovnik, Croatia, during June 25–27, 2012. The conference was organized by the University of Zagreb, Faculty of Electrical Engineering and Computing, KES International and its Focus Group on Agent and Multi-agent Systems. The KES-AMSTA conference is a subseries of the KES conference series.

Following the successes of previous KES Symposia on Agent and Multi-Agent Systems – Technologies and Applications, held in Wroclaw, Poland (KES- AMSTA 2007), Incheon, Korea (KES-AMSTA 2008), Uppsala, Sweden (KES-AMSTA 2009) and Gdynia, Poland (KES-AMSTA 2010), it was decided to run KES-AMSTA 2011 in Manchester, UK, as a full conference. KES-AMSTA 2012 featured the usual keynote talks, oral presentations and invited sessions closely aligned to the established themes of the conference.

The aim of the conference was to provide an internationally respected forum for scientific research in the technologies and applications of agent and multi-agent systems. This field is concerned with the development and evaluation of sophisticated, AI-based problem-solving and control architectures for both single-agent and multi-agent systems. Current topics of research in the field include (amongst others) agent-oriented software engineering, beliefs, desires and intentions, agent co-operation, co-ordination, negotiation, organization and communication, distributed problem solving, specification of agent communication languages, formalization of ontologies and conversational agents. Special attention is paid to the feature topics: Intelligent technologies and applications in the area of e-health, social networking, self-organizing systems, economics and trust management.

The conference attracted a substantial number of researchers and practitioners from all over the world who submitted their papers for ten main tracks covering the methodology and applications of agent and multi-agent systems, one workshop and five special sessions on specific topics within the field.

Submissions came from 20 countries. Each paper was peer reviewed by at least two members of the International Program Committee and International Reviewer Board. In all, 66 papers were selected for oral presentation and publication in the proceedings volume of KES-AMSTA 2012.

The Program Committee defined the following main tracks: Knowledge and Learning Agents, Virtual Organizations, Business Processing Agents, Multi-agent Systems, Mental and Holonic Models, Self-Organization, Conversational Agents and Agent Teams, and Multi-agent Systems in Distributed Environments.

In addition to the main tracks of the conference the First International Workshop on Trustworthy Multi-Agent Systems (TRUMAS 2012) and the following five special sessions were also hosted: Intelligent Workflow, Cloud Computing and Intelligent Systems; Digital Economy; Assessment Methodologies in Multi-agent and Other Paradigms; ICT-Based Alternative and Augmentative Communication, as well as a Doctoral Track. This year's conference also organized a Work in Progress Track as a half-day workshop taking place on the final day of the conference, giving early career stage researchers the opportunity to present their work.

Accepted and presented papers highlight new trends and challenges in agent and multi-agent research. We hope that these results will be of value to the research community working in the fields of artificial intelligence, collective computational intelligence, robotics, dialogue systems and, in particular, agent and multi-agent systems, technologies and applications.

We would like to express our sincere thanks to the Honorary Chair Ignac Lovrek, University of Zagreb, Croatia, for his support and help.

The Chairs' special thanks go to Tanja Grzilo, University of Zagreb, CAAS Dubrovnik, Croatia, and to the Local Organizing Chair Igor Cavrak, for his excellent work. Thanks are due to the Program Co-chairs, all Program and Reviewer Committee members and all the additional reviewers for their valuable efforts in the review process, which helped us to guarantee the highest quality of selected papers for the conference.

We would like to thank our main sponsor, University of Zagreb. Our special thanks also go to Springer for publishing the proceedings.

We would also like to express our thanks to the keynote speakers, Michael Luck, King's College London, UK, Wolfgang Ketter, Erasmus University, The Netherlands, and Kimon Valavanis, University of Denver, USA, for their interesting and informative talks of world-class standard. We cordially thank all of the authors for their valuable contributions and all of the other participants in this conference. The conference would not have been possible without their support.

April 2012

Gordan Jezic
Mario Kusek
Ngoc Thanh Nguyen
Robert J. Howlett
Lakhmi C. Jain

Organization

KES-AMSTA 2012 was organized by KES International – Innovation in Knowledge-Based and Intelligent Engineering Systems—and the University of Zagreb, Faculty of Electrical Engineering and Computing.

Honorary Chairs

I. Lovrek	University of Zagreb, Croatia
L. C. Jain	University of South Australia

General Co-chairs

G. Jezic	University of Zagreb, Croatia
N.T. Nguyen	Wroclaw University of Technology, Poland

Executive Chair

R.J. Howlett	University of Bournemouth, UK

Program Co-chairs

M. Kusek	University of Zagreb, Croatia
R. Katarzyniak	Wroclaw University of Technology, Poland
D. Trcek	University of Ljubljana, Slovenia

Local Organizing Chair

I. Cavrak	University of Zagreb, Croatia

Invited Session Chair

D. Jevtic	University of Zagreb, Croatia

Doctoral Track Chair

V. Podobnik	University of Zagreb, Croatia

Publicity Co-chairs

A. Petric	University of Zagreb, Croatia
I. Bojic	University of Zagreb, Croatia

KES-AMSTA Symposium Series and Focus Group on Agent and Multi-agent Systems Chair

N.T. Nguyen	Wroclaw University of Technology, Poland

KES International Conference Series Chairs

R.J. Howlett	University of Bournemouth, UK
Lakhmi Jain	University of South Australia, Australia

Keynote Speakers

Michael Luck
King's College London, UK
Behavior Regulation and Normative Systems

Wolfgang Ketter
Erasmus University, The Netherlands
Competitive Simulations: Lessons Learned from the Trading Agent Competition

Kimon Valavanis
University of Denver (DU), USA
Challenges in Unmanned Systems Swarms
A Wireless, Multi-agent Distributed System Perspective

International Program Committee

Ahmad Taher Azar	IGI Global, USA
Marina Babic Babac	University of Zagreb, Croatia
Dariusz Barbucha	Gdynia Maritime University, Poland
Stjepan Bogdan	University of Zagreb, Croatia
Andrej Brodnik	University of Ljubljana, Slovenia
Frantisek Capkovic	Slovak Academy of Sciences, Slovakia
Angela Consoli	KES Centre, University of South Australia
Keeley Crockett	Manchester Metropolitan University, UK
Ireneusz Czarnowski	Gdynia Maritime University, Poland
Trong Hai Duong	Inha University, Korea
Konrad Fuks	Poznañ University of Economics, Poland
Anne Håkansson	KTH Royal Institute of Technology, Sweden
Chihab Hanachi	University of Toulouse, France

Ronald L. Hartung	Franklin University, USA
Quang Hoang	Hue University, Vietnam
Zeljko Hocenski	University J.J. Strosssmayer of Osijek, Croatia
Mirjana Ivanovic	University of Novi Sad, Serbia
Dragan Jevtic	University of Zagreb, Croatia
Jason J. Jung	Yengnam University, Korea
Radoslaw Katarzyniak	Wroclaw University of Technology, Poland
Branko Kavsek	Josef Stefan Institute, Slovenia
Dariusz Krol	Wroclaw University of Technology, Poland
Adrianna Kozierkiewicz-Hetmańska	Wroclaw University of Technology, Poland
Marin Lujak	University Rey Juan Carlos, Spain
Ngoc Thanh Nguyen	Wroclaw University of Technology, Poland
James O'Shea	Manchester Metropolitan University, UK
Marin Orlic	University of Zagreb, Croatia
Nedjeljko Peric	University of Zagreb, Croatia
Vedran Podobnik	University of Zagreb, Croatia
Radu-Emil Precup	Politehnica University of Timisoara, Romania
Rajesh Reghunadhan	Network Systems and Technologies Ltd., India
Darko Stipanicev	University of Split, Croatia
Armin Stranjak	Rolls-Royce, UK
Zbigniew Telec	Wroclaw University of Technology, Poland
Johnson Tran	Weapons Systems Division, Australia
Bogdan Trawinski	Wroclaw University of Technology, Poland
Krunoslav Trzec	Ericsson Nikola Tesla, Croatia
Taketoshi Ushiama	Kyushu University, Japan
Bay Vo	Ho Chi Minh City University of Technology, Vietnam
Toyohide Watanabe	Nagoya University, Japan
Mario Žagar	University of Zagreb, Croatia

Workshop and Invited Session Chairs

Intelligent Workflow, Cloud Computing and Intelligent Systems

Jessica Chen-Burger	University of Edinburgh, Scotland, UK
Lakhmi Jain	University of South Australia, Australia
Fang-Pang Lin	National Centre for High-Performance Computing, HsinChu, Taiwan

Digital Economy

Arkadiusz Kawa	Poznan, University of Economics, Poland
Jessica Chen-Burger	University of Edinburgh, Scotland, UK
Konrad Fuks	Poznan, University of Economics, Poland

Assessment Methodologies in Multi-agent and other Paradigms

Jadranka Sunde	Defence Science and Technology Organisation and University of South Australia
Marina Cicinsain	University of Rijeka, Croatia

TRUMAS 2012 (First International Workshop on Trustworthy Multi-agent Systems)

Nicola Dragoni	Technical University of Denmark, Denmark
Manuel Mazzara	Newcastle University, UK

ICT-Based Alternative and Augmentative Communication

Zeljka Car	University of Zagreb, Croatia

Sponsoring Institutions

Ministry of Science, Education and Sports of the Republic of Croatia
University of Zagreb, Faculty of Electrical Engineering and Computing

Table of Contents

Keynote Lectures

Virtual Organizations

Knowledge and Learning Agents

Knowledge Agents

Intelligent Workflow, Cloud Computing and Intelligent Systems

Self-organization

ICT-Based Alternative and Augmentative Communication

Multi-Agent Systems I

Multi-Agent Systems II

Mental and Holonic Models

Assessment Methodologies in Multi-agent and other Paradigms

Business Processing Agents

Doctoral Track

TRUMAS 2012 (First International Workshop on Trustworthy Multi Agent Systems)

Conversational Agents and Agent Teams

Digital Economy

Multi-Agent Systems in Distributed Environments

Competitive Simulations: Lessons Learned from the Trading Agent Competition

Wolfgang Ketter

Erasmus University, Netherlands
WKetter@rsm.nl

Abstract. Many important developments in artificial intelligence have been stimulated by organized competitions that tackle interesting, difficult challenge problems, such as chess, robot soccer, poker, robot navigation, stock trading, and others. Economics and artificial intelligence share a strong focus on rational behavior. Yet the real-time demands of many domains do not lend themselves to traditional assumptions of rationality. This is the case in many trading environments, where self-interested entities need to operate subject to limited time and information. With the web mediating an ever broader range of transactions and opening the door for participants to concurrently trade across multiple markets, there is a growing need for technologies that empower participants to rapidly evaluate very large numbers of alternatives in the face of constantly changing market conditions. AI and machine-learning techniques, including neural networks and genetic algorithms, are already routinely used in support of automated trading scenarios. Yet, the deployment of these technologies remains limited, and their proprietary nature precludes the type of open benchmarking that is critical for further scientific progress.

The Trading Agent Competition was conceived to provide a platform for study of agent behavior in competitive economic environments. Research teams from around the world develop agents for these environments. During annual competitions, they are tested against each other in simulated market environments. Results can be mined for information on agent behaviors, and their effects on agent performance, market conditions, and the performance and behavior of competing agents. After each competition, competing agents are made available for offline research. We will discuss results from various competitions (Travel, Supply-Chain Management, Market Design, Sponsored Search, and Power Markets).

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, p. 1, 2012.

Behaviour Regulation and Normative Systems

Michael Luck

King's College London, UK
michael.luck@kcl.ac.uk

Abstract. Cooperation is the fundamental underpinning of multi-agent systems, allowing agents to interact to achieve their goals. However, agents must manage the risk associated with interacting with others who have different objectives, or who may fail to fulfill their commitments. There are many ways in which such a desirable social order may be encouraged or even mandated. For example, trust offers a mechanism for modeling and reasoning about reliability, honesty, etc., while organisations and norms provide a framework within which to apply them, and motivations provide a means for representing and reasoning about overall objectives. In this talk, I will consider the role of all these aspects, with a particular focus on norms in regulating behaviour.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, p. 2, 2012.

Challenges in Unmanned Systems Swarms: A Wireless, Multi-agent Distributed System Perspective

Kimon Valavanis

University of Denver (DU), USA
Kimon.Valavanis@exch.du.edu

Abstract. Networked swarms may be considered as heterogeneous, distributed agent-based systems (or distributed wireless sensor networks), where the concept/definition of an 'agent' may be interpreted in the most general sense. This is true for networked swarms of unmanned systems, as well as manned-unmanned systems, where the term 'agent' may be thought of as (an individual) 'asset'. When considering swarms in (fixed or loose) 2-D/3-D formations, each swarm may be also thought of as a 'group of agents' with initially specific, but dynamically and in real-time modifiable goals pertaining to an assigned mission or missions. As such, when studying such swarms, there are major challenges related to navigation and control, communications (including secured inter- and intra-swarm communication), coordination and collaboration, robustness and fault tolerance, security, to name just a few.

In this seminar, the 'connection' between the swarm formation consideration and the 'distributed (agent-based) sensor network' consideration will be established. Assets will be considered as sensor nodes, and the overall swarm as a sensor network. A well integrated methodology will be presented, in which on top of navigation/control aspects, communication and networking aspects as well as software security will be discussed. The swarm formation will be dictated / modified based on communication/data transmission requirements, in addition to control requirements. Further, it will be shown how networked swarms / distributed agent-based systems may be organized in architectures comprised of swarms, each containing a 'swarm head' as leader with all 'swarm heads' being connected via a backbone network.

Case studies of aerial, ground and aerial-ground unmanned vehicle swarms will be demonstrated experimentally.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, p. 3, 2012.

On Mobile Target Allocation with Incomplete Information in Defensive Environments

Marin Lujak, Holger Billhardt, and Sascha Ossowski

University Rey Juan Carlos, C/Tulipán s/n, 28933 Móstoles, Madrid, Spain

Abstract. In this paper, the decentralized mobile target allocation problem is researched. We assume the existence of two groups of mobile agents: attackers and targets. Every attacker agent gets allocated to its best target based on the communication and coordination with the rest of the group positioned within a limited communication range (radius) and moves towards it. This is performed through the dynamic iterative auction algorithm with mobility without any insight in the decision-making processes of the other agents. Targets are mobile and combine two strategies to escape from the attacking group: moving linearly and randomly away from the attacker. We explore the dynamics of the allocation solution in respect to the mentioned escape strategies, maximum step size in each run (velocity), and the agents' varying communication range when the latter is not sufficient to maintain a connected communication graph.

1 Introduction

The efficiency of resource utilization in a dynamic multi-agent target allocation is one of the essential factors of the mission's success since it implies the minimization of the cost and the time of mission execution. In unstable dynamic environments agents need to share adequate amounts of information and to mutually coordinate in order to allocate targets without conflicts. Furthermore, they need to update their locally limited information, and respond to true changes in the environment in an autonomous way. The failure to do so may result in a phenomenon called churning, whereby agents constantly replan based on their latest noisy information, and oscillate between targets without ever reaching anyone of them. This is why decentralized algorithms that can adequately correspond to the changing communication network topologies are an important topic of the collaborative target allocation research (see, e.g., [16, 20]).

A multi-agent system in which agents have limited communication capabilities may achieve an optimal target assignment solution if the global information is up- to-date and at the disposal of all decision makers (agents) at least in a multi-hop fashion; sufficient condition for optimal solution is that the communication graph among agents is connected [20, 21]. The topology of the communication graph depends on the choice of the agents' transmitting range (radius): the larger the range, the less likely it is that the network becomes disconnected. From a communication-theoretic perspective, the critical transmitting range (CTR) is a minimum common value of the agents' transmitting range that produces a connected communication graph. If the communication range is under CTR, then we naturally expect to achieve a poor group decision making and lower levels of cooperation and hence assignment performance.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 4–13, 2012.

In [13], we started the research on multi-agent target allocation with a disconnected communication graph and considered a neutral environment with static targets. Each agent of a target group was, as in this paper, supposed to be assigned to at most one mobile agent of the opposite group. We proposed the decentralized iterative auction algorithm with mobility for the scenario with disconnected communication network. The presented version of the auction algorithm was shown through simulations to be resistent to communication network failures and to adapt to local incomplete information. In [14] we went a step forward and examined the strictly collaborative case where both groups want to get allocated to one another and when assigned, move closer until they reach one another.

In continuation of the former work, this paper investigates the problem of a dynamic and decentralized multi-agent mobile target allocation with locally incomplete information in a dynamic environment with escaping targets. In this scenario, the attacker group has the objective of a global minimization of the total distance of movement towards the target group while targets combine two different strategies of movement: running away from attacker in a linear or a random fashion. We examine the influence of the communication range under the CTR on the effectiveness of allocation and reaching the targets.

Assuming the same velocity for both groups, the best individual strategy for a target is to move linearly away from the closest attackers. An interesting simulation result is that, a coordinated escaping movement of all the members of the target group in the same direction will not have much different global result (approximately 10 percent of all the targets are caught by the attacker) from the pure linear escaping strategy. This might help in battlefield scenarios where the reinforcement base of the target group is located at some specific geographical position.

The paper is organized as follows. In Section 2 we tackle related work. In Section 3, problem formulation and definitions are presented. In Section 4, the modified distributed auction algorithm for the case of mobile agents and their targets and disconnected communication graph is presented. Section 5 contains simulation results demonstrating the performance of the presented auction algorithm. The conclusions are found in Section 6.

2 Related Work

The desired level of autonomy in which each agent can contribute to the overall mission objective independently can only be accomplished when agents create their own plans while exchanging important information and coordinating with others. This can be done through a market-based coordination approach which has been implemented in many multi-agent systems (e.g., [4, 6–9, 12, 13, 15, 21]).

Dynamic task reallocation allows mobile agents to change their behavior in response to environmental changes or actions of other agents in order to improve overall system performance. There are a number of task allocation algorithms that rely on the assumption of consistent information requiring a complete communication graph among agents (e.g., [1–3, 9–11, 15, 20, 21]).

A multi-agent system in which agents have limited communication capabilities achieves an optimal target assignment solution if the global information is up- to-date and at the disposal of all decision makers (agents) at least in a multi-hop fashion. A sufficient condition for an optimal solution is that the communication graph among agents is connected [20, 21]. At CTR, all global data is available to each agent and if the communication range is lower than CTR, agents' decisions are sub-optimal due to the lack of updated information. Critical transmitting range was a topic of diverse works, e.g., [5, 17–19].

Within the defensive environment context, the correlation between the dynamics of the target allocation in respect to the variation of the agents' communication range under the critical transmitting range and escaping strategy of targets has not been explored so far to the best of our knowledge.

3 Problem Formulation and Definitions

Considering a time horizon made of T time periods, given are two distinct groups, each one made of n agents: attacker group $A = \{a_1, \ldots, a_n\}$ and target group $\Theta = \{\theta_1, \ldots, \theta_n\}$, both represented by points in the plane. Agents initial coordinates are positioned, w.l.o.g., in a square environment $E = [0, l]^2 \subset \mathbf{R}^2$ of side length $l > 0$, with $p_a(t) \in E$ being the position of agent $a \in A$ at the beginning of each time period $t = 1, \ldots, T$, and $q_\theta(t) \in E$ being the position of agent $\theta \in \Theta$ at time $t = 1, \ldots, T$.

Each agent, $a \in A$, is an autonomous and independent decision maker which is described by the touple

$$a = \{p_a(t),\ \rho_a,\ d_{max}^{[a]}\} \ , \tag{1}$$

and, similarly, agent $\theta \in \Theta$ by the touple

$$\theta = \{q_\theta(t),\ \rho_\theta,\ d_{max}^{[\theta]}\} \ , \tag{2}$$

where $\rho_a \in \mathrm{R}_{>0}$ and $\rho_\theta \in \mathrm{R}_{>0}$ are a fixed transmitting (communication) range (radius) of agent a's and agent θ's wireless transceiver for limited range communication respectively. This implies that each agent can have a possibly different local information (i.e., situational awareness (SA)) depending on the value of ρ_a and ρ_θ. Furthermore, $d_{max}^{[a]}$ and $d_{max}^{[\theta]}$ are the maximum movement distance (maximum step size) of agent $a \in A$ and agent $\theta \in \Theta$ respectively, in each time period t. At any time period t, each agent a knows its position $p_a(t)$ and the position $q_\theta(t)$ of each opposite group's agent $\theta \in \Theta$. Similarly, the same holds for agents $\theta \in \Theta$. Let $c_{a\theta}(t)$ be the function which represents the (Euclidean) distance between the positions of agents a and θ.

Examples of such a two-group agent setup can be found in mobile sensor networks and in unmanned aerial vehicles (UAVs) where mobile agents can move to follow and reach mobile targets, the position of which is globally known to all the agents (through, e.g., global positioning system).

As the attacking agents move in the environment towards their target agents, the conditions of the relative proximity of the agents of two groups in the environment change so that the closest agent $a(\theta)$ in one period can become more distant than some other agent which was further away in the period before.

In each period t, each agent a is able to communicate to a set of agents $C_a(t) \subseteq A$, (belonging to the same connected component of its own group) reachable in a multi-hop fashion within the communication graph at time period t, the latter is a random geometric graph (RGG) [5], that is the undirected graph $G_a(t) = (A, E_a(t))$ with vertex set A randomly distributed in some subset of R^2, and edge set $E_a(t)$ with edge $(i, j) \in E_a(t)$ if and only if $\|p_i(t) - p_j(t)\|_2 \leq \rho_a$.

The similar logic applies to the agents of the opposite group Θ which are able to communicate to a set of agents $C_\theta(t) \subseteq \Theta$ reachable in a multi-hop fashion within the communication graph $G_\theta(t) = (A, E_\theta(t))$. In this way, two agents of a same group which are not within the communication range of each other can communicate over a third agent within the same group (communication relay point) in a multi-hop fashion as long as the latter is placed within the communication range of the both. Therefore, agent a together with the set of agents communicating with the same induce a connected subgraph (connected component) of $G_a(t)$. The same principle stands for agent $\theta \in \Theta$ and its communicating agents resulting in a connected component of $G_\theta(t)$.

We consider the problem of dynamic assignment of attacker agents to mobile targets: A to Θ, where each attacker agent has to be assigned to at most one mobile target agent. On the other hand, agents belonging to Θ have to escape attacking agents. The total traveled distance of all attackers which move towards their assigned targets has to be dynamically minimized while the targets maximize the distance in moving away from attackers. We assume that no a priori global assignment information is available and that agents are collaborative within their group and only receive information through their local interaction with the environment and with the connected agents in the communication graph of the same group. Agents communicate the assignment data and negotiate with the agents of the same group while there is no direct communication of the agents of opposite groups. T is the upper bound on the number of time periods in which all the agents reach their distinct assigned targets.

4 Dynamic Online Auction with Mobility

Each attacking agent has its own copy of the dynamic iterative auction algorithm with mobility which serves as a controller of its actions as it moves toward its assigned target in each period. The algorithm was described in [14] and inspired by the Bertsekas auction algorithm [2]. In the following we shortly present its working principle.

Each agent a keeps in its memory its most recent knowledge about the actual bidding value of every target $\theta \in \Theta$, and the set S_a of its most recent knowledge about all the agents' assignments. None of the local information has to coincide with the actual target value and agents' assignments, respectively; they may also differ from one agent to another due to the dynamics of their previous communication and local interaction with the other agents and targets in the environment.

Each target can be reached or free depending if there is already an attacker agent present on its position. Initially, each attacker agent's local copy of assignments $S_a(t = 1, h = 0)$ is assumed empty and all target values $v_{\theta,a}(1, 0)$ are set to zero. At the beginning of each time period $t \in [1, \ldots, T]$, a new round of iterative auction is performed, starting from the assignments and target values locally available to the agents from the end of the round before. Each agent performs following steps:

- broadcasts within its connected component its identity, the assigned target (if any) and its status on that target (whether it has arrived or not to the target),
- sends and receives all target values $v_{\theta,a}(t, h-1)$ and assignments $S_a(t, h-1)$ for all $\theta \in \Theta$ from/to the agents within the connected component $C_a(t)$ agent a belongs to, and
- updates its local list of the assignments $S_a(t, h)$ and target values $v_{\theta,a}(t, h)$ by adopting the largest value $v_\theta(t, h-1)$ among the set of agents $C_a(t)$ within the connected component for each target of interest $\theta \in \Theta$ and the assignment resulting from this value.

However, if the target θ_a is assigned to more than one agent in $C_a(t)$, its assignment is canceled making it unassigned and eligible for bidding in the ongoing round unless if some agent has already come to its target position q_{θ_a}; in that case, the agent on the target position remains assigned to the target while other agents within the connected component update the value of the target θ_a and cancel the assignment to the same.

- Next, all agents scan the environment for closer targets than their assigned ones; if one is found, they break the assignment with their assigned target and become eligible for biding in the ongoing round.
- If agent a is unassigned, it finds its best target, calculates the bid value and bids for that target using the bidding and assignment procedure described in [14].

When all agents are assigned to their respective targets in the final auction iteration, say $h_{fin}(t)$, at round t of the iterative auction, the agent movement phase takes place:

- if agent $a \in A$ is not at its target position p_{θ_a}, it moves one step toward θ_a, covering at most a maximum distance $d_{max}^{[a]}$.
- Once when agent a comes to the position of its assigned and available target, it changes its target value $v_{\theta_a a}(t, h_{fin}(t))$ to $-\inf$, and broadcasts the latter to the agents in $C_a(t)$.

The presented algorithm works also when the number of agents differs from the number of targets. In fact, if the number of targets is not less than the number of agents, the assignment process terminates when each agent reaches its assigned target; otherwise, the assignment process terminates when all the unassigned agents realize that all the targets are occupied.

5 Simulation Setup and Results

We simulate a two-group Multi-Agent System with previously described auction algorithm implemented in MatLab and examine the assignment solution in respect to the communication range, the maximum distance of movement in each time period, and targets' escaping strategies. Principally, three different targets strategies are examined:

- moving linearly away in the direction opposite of the closest attackers,
- moving away in a coordinated manner in a group-wide agreed direction on a straight line, and
- moving away in a coordinated manner in agreed direction with individually chosen random factor varying from 0 up to 90, 180, 270, and 360 degrees of the intended course.

The last strategy means that targets, once when agreed on the angle of movement α, can individually vary their moves within the range of the random factor, i.e., if, e.g., a random factor is 90 degrees, then they can choose to move in any direction $\alpha \pm 45°$. This implies that, with 90 degrees randomness, the group will, on average, move in the desired direction while with the randomness of 360 degrees, the group will on average, move around their initial positions.

Ten experiments are performed for each of the targets' escaping strategies for 60 agents (30 attackers and 30 targets) with their initial coordinates positioned uniformly randomly in $[0, 50]^2 \subset \mathbf{R^2}$. Maximum step size d_{max} varies from 1 to 30 while the value of communication range ρ is set from 0 to 18, the latter guaranteeing initially the full communication network.

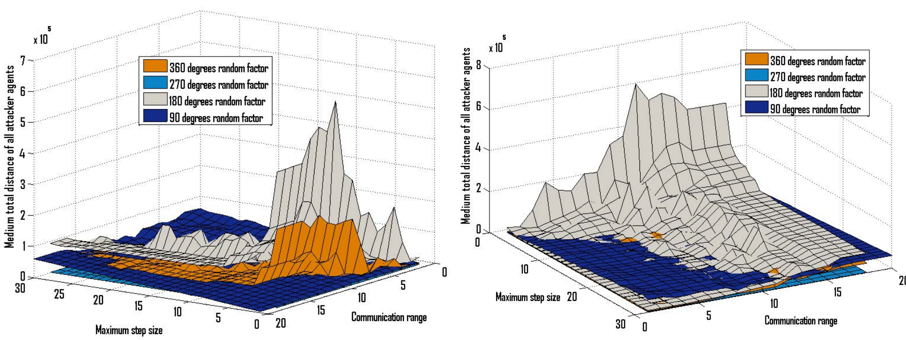


Fig. 1. Two perspectives of medium total distance crossed by attacker agents in respect to communication range ρ and maximum step size d_{max}

We assume maximum distances crossed in each time period to be equal for the two groups, that is, $d_{max}^{[a]} = d_{max}^{[\theta]} = d_{max}$; the same assumption applies to the communication range of the two groups, i.e., $\rho_a = \rho_\theta = \rho$, $\forall a \in A$ and $\forall \theta \in \Theta$. In such a context, the targets' strategy of escaping in linear motion away from attackers results in the most efficient run-away strategy in all experimented random instances. This individual random movement might imply that targets as a group might dissolve. To keep the group together, it might be efficient to try escaping the attackers moving in a previously agreed direction. An interesting simulation result is that, a coordinated escaping movement of all the members of the target group in the same direction will not have much different global result (approximately 10 percent of all the targets are caught by the attackers) from the pure linear escaping strategy. This strategy keeps the group together,

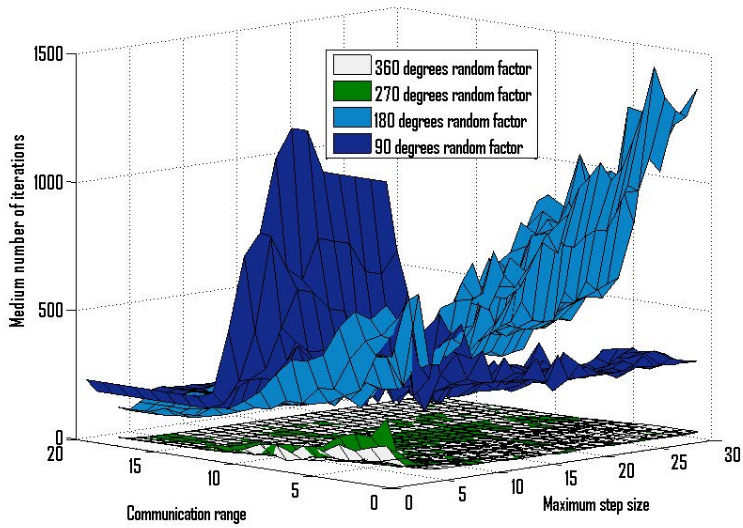


Fig. 2. First perspective of medium total number of algorithm's iterations until all targets are reached

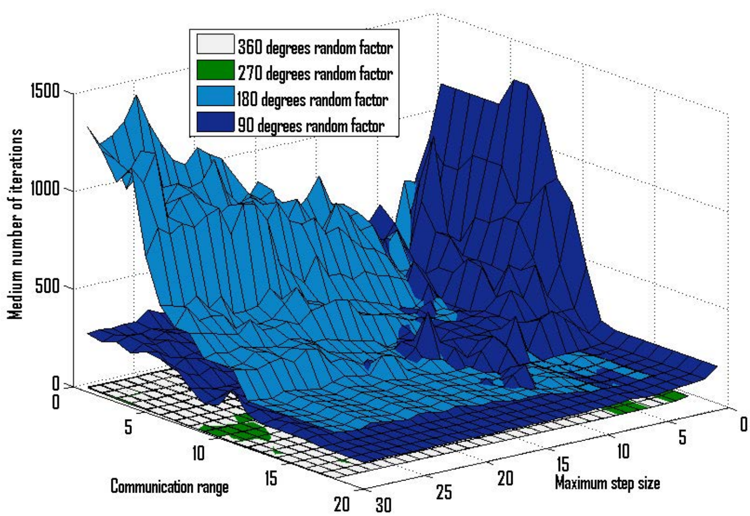


Fig. 3. Second perspective of medium total number of algorithm's iterations until all targets are reached

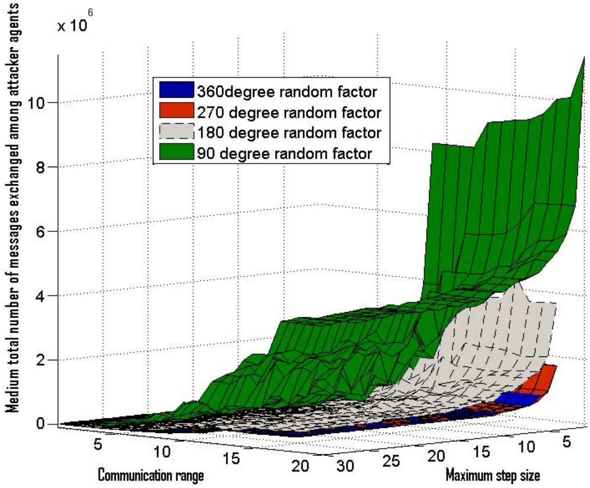


Fig. 4. Medium total number of messages exchanged among attacker agents through the execution of the algorithm

guaranteeing to most of targets, in the hostile environment, the possibility of reaching their base station (if any). This strategy might help in battlefield scenarios where the reinforcement base of the target group is located at some specific geographical position.

The attackers' possibility of reaching the targets highly improves when targets include random factor in their movement as is seen in Figures 1, 2, and 3. Please note that Figures 2, and 3 are for better visibility two perspectives of the same graph. The same holds for the left and the right part of Figure 1. Already at the random factor of 90 degrees, all the targets are reached at some point by attackers. The average assignment solution (over 10 instances) for the target strategies with random factors is presented in Figure 1. The latter represents the dynamics of the total distance crossed by attackers until reaching targets in respect to varying maximum step size d_{max} and the communication range ρ. Figures 2 and 3 present the medium total number of runs of the algorithm with targets escape strategies including random factor. Please note that here the number of algorithm's iterations is the one until the last agent reaches its target. The peaks in the graph are due to this specificity of the algorithm. It might be the case that most of the agents reach their targets quite soon but that one target escapes and is followed by the last unallocated agent. This is why the maximum (peak) distances do not have to necessarily respond to the maximum (peak) number of iterations. Furthermore, Figure 4 presents a medium number of messages exchanged among attackers during the total running time of the algorithm. As seen from the Figures, the presented auction algorithm with included mobility is stable in those scenarios, i.e., it produces a feasible allocation and target reaching solution.

Moreover, the lower the random factor in the movement of targets, the more likely targets are to escape from attackers when the communication network is disconnected with high probability (Figure 1). This fact also implies a higher number of iterations, and obviously a higher distance, in average, passed by targets before getting caught by attackers. The allocation situation is not that straightforward in the case with a higher communication range that guarantees a connected communication network with high probability. In fact, the strategy with random factor of 180 degrees for the communication range $\rho > 12$ appears as the one with the longest traveled distance until targets are finally reached.

Relation of the targets strategies with the allocation result is also influenced by the maximum step size, i.e., velocity. The higher the maximum step size, the less evident is the superiority for targets of the strategies with lower random factor. On the contrary, in Figure 3, it is evident that the strategy with 180 degree random factor results in the highest number of iterations starting from the maximum step size $d_{max} = 20$. The number of messages exchanged in the attacker's group grows as the random factor in the targets movement decreases since the reaching of targets process takes more iterations. For the information regarding the complexity of the iterative auction algorithm with mobility, see, [14].

6 Conclusions

In this paper, we explored how negotiation through iterative auction algorithm with mobility influences target allocation solution when the agents' communication range is insufficient for connected communication graph. Furthermore, we examined the dynamics of the assignment solution in respect to the maximum distance stepped by the agents between two consecutive time periods and some escape strategies for targets. Targets applied six different escaping strategies, including moving linearly away from the closest attackers, moving away in the same direction on the straight line, and with some variation within 90, 180, 270 and 360 degrees of a predetermined course.

The scenarios presented in this paper address important aspects of the UAV and rescue target allocation problem. We evaluated the performance of the proposed algorithm with varying range of communication and velocity (step size in each iteration) and demonstrated its performance and robustness.

The results raise many interesting questions for future study. For example, we have assumed that targets locations are known at the start of the mission. It is much more likely that in real mission conditions, an attacker (or rescue) team has an approximate idea about some and not all the targets which are discovered as the mission evolves. Since the allocation solution is influenced by uncertainty due to incompleteness of information, for the future work we consider more formal analysis of mobile target allocation, its efficiency and robustness.

Acknowledgements. This work was supported in part by the Spanish Ministry of Science and Innovation through the projects "AT" (Grant CONSOLIDER CSD2007-0022, INGENIO 2010) and "OVAMAH" (Grant TIN2009-13839-C03-02).

References

1. Alighanbari, M., How, J.P.: Robust decentralized task assignment for cooperative UAVs. In: Proc. of AIAA Conf. on Guidance, Navigation and Control (2006)
2. Bertsekas, D.P.: Auction algorithms for network flow problems: A tutorial introduction. Comput. Opt. and Applications 1(1), 7–66 (1992)
3. Choi, H.L., Brunet, L., How, J.P.: Consensus-based decentralized auctions for robust task allocation. IEEE Transactions on Robotics 25(4), 912–926 (2009)
4. Dias, M.B., Zlot, R., Kalra, N., Stentz, A.: Market-based multirobot coordination: A survey and analysis. Proc. of IEEE, Spec. Iss. on Multirob. Coord. 94(7), 1257–1270 (2006)
5. Díaz, J., Mitsche, D., et al.: On the connectivity of dynamic random geometric graphs. In: Proc. of 19th ACM-SIAM Symposium on Discrete Algorithms, pp. 601–610 (2008)
6. Elango, M., Nachiappan, S., Tiwari, M.K.: Balancing task allocation in multi-robot systems using k means clustering and auction based mechanisms. Expert Systems With Applications 38(6), 6486–6491 (2010)
7. Gerkey, B.P., Matarić, M.J.: A formal analysis and taxonomy of task allocation in multi-robot systems. The Intern. Journ. of Rob. Research 23(9), 939–954 (2004)
8. Gerkey, B.P., Matarić, M.J.: Multi-robot task allocation: Analyzing the complexity and optimality of key architectures. In: Proc. of ICRA 2003, vol. 3, pp. 3862–3868 (2003)
9. Gerkey, B.P., Matarić, M.J.: Sold!: Auction methods for multi-robot control. Proc. IEEE Trans. on Robotics and Automat. 18(5), 758–768 (2002)
10. Gil, A.E., Passino, K.M., Ganapathy, S., Sparks, A.: Cooperative scheduling of tasks for networked uninhabited autonomous vehicles. In: Proc. of 42nd IEEE Conf. on Decision and Control, pp. 522–527 (2003)
11. Hoeing, M., Dasgupta, P., Petrov, P., O'Hara, S.: Auction-based multi-robot task allocation in comstar. In: Proc. of the 6th Inter. Joint Conf. on Auton. Agents and Multiagent Systems, pp. 1–8 (2007)
12. Lerman, K., Jones, C., Galstyan, A., Matarić, M.J.: Analysis of dynamic task allocation in multi-robot systems. The Intern. Journal of Robotics Research 25(3), 225–253 (2006)
13. Lujak, M., Giordani, S.: On the Communication Range in Auction-Based Multi-Agent Target Assignment. In: Bettstetter, C., Gershenson, C. (eds.) IWSOS 2011. LNCS, vol. 6557, pp. 32–43. Springer, Heidelberg (2011)
14. Lujak, M., Giordani, S., Ossowski, S.: Value of Incomplete Information in Mobile Target Allocation. In: Klügl, F., Ossowski, S. (eds.) MATES 2011. LNCS (LNAI), vol. 6973, pp. 89–100. Springer, Heidelberg (2011)
15. Nanjanath, M., Gini, M.: Repeated auctions for robust task execution by a robot team. Robotics and Autonomous Sys. 58(7), 900–909 (2010)
16. Ponda, S., Redding, J., Choi, H.L., How, J.P., Vavrina, M., Vian, J.: Decentralized planning for complex missions with dynamic communication constraints. In: American Control Conference (ACC), pp. 3998–4003 (2010)
17. Penrose, M.: Random geometric graphs. Oxford Univ. Press, USA (2003)
18. Santi, P.: The critical transmitting range for connectivity in mobile ad hoc networks. IEEE Trans. on Mob. Comp. 4(3), 310–317 (2005)
19. Santi, P., Blough, D.M.: An evaluation of connectivity in mobile wireless ad hoc networks. In: Proc. of Int. Conf. on Depend. Sys. and Netw., pp. 89–98 (2002)
20. Savkin, A.V.: The problem of coordination and consensus achievement in groups of autonomous mobile robots with limited communication. Nonlinear Analysis 65(5), 1094–1102 (2006)
21. Zavlanos, M.M., Spesivtsev, L., Pappas, G.J.: A distributed auction algorithm for the assignment problem. In: Proc. of 47th IEEE Conf. on Dec. and Contr., pp. 1212–1217 (2008)

Bayesian Proprioceptor for Forest Fire Observer Network

Ljiljana Šerić, Maja Štula, Darko Stipaničev, and Maja Braović

Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture,
University of Split, R. Boškovića 32, 21000 Split, Croatia
ljiljana@fesb.hr
http://laris.fesb.hr

Abstract. The paper describes design and implementation of Bayesian proprioceptor for forest fire observer network. The proprioceptor, sometimes also referred to as network observer has task of syintactical and semantical sensor and data validation in advanced sensor network Multi agent Bayesian network is used for cooperative data analysis and data understanding having false alarm reduction as final goal. A multi agent system for data sampling and data analysis is described. The proprioceptor is deployed as a part of intelligent forest fire monitoring system (iForestFire).

Keywords: Probabilistic reasoning, observer network, proprioceptor.

1 Introduction

Whether early forest fire detection is performed using satellite images, infrared, near infrared or visual spectrum camera, high false alarm rate is always the most important problem [1]. Large number of false alarms is the main reason why forest fire monitoring system can not operate as a fully automatic system and still needs to be semi automatic. On the other hand, majority of false alarms are easily detected by human operator using simple human reasoning.

Forest fire detection can be performed using only video image, but more accurate results are obtained when meteorological data is also introduced. Meteorological data is than used in alarm post processing as additional data about environment. iForestFire system utilizes observer network [4] for forest fire detection, where video image and additional sensors data is fused into scenario of environment making artificial perception system for forest fire perception.

Perception in observer network is done in two steps – proprioception with main task of sensor data validation performed by network observer and exterioception or creation of external scenario description performed by phenomenon observer.

Probabilistic reasoning has shown good results when dealing with uncertainty [5,6,7,8], and thus can be one of possible approaches in sensor data validation with final goal of false alarm reduction.

In this paper we will show how probabilistic causal model can be used for false alarm reduction by introducing a measure called weight of soundness (WOS) assigned

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 14–23, 2012.

to a set of measurements taken at the same time cycle. WOS measure with defined threshold value is used as a classifier for false forest fire alarm.

In the next section observer network and iForestFire will be described to the extent necessary for this paper. The following section will be divided in two parts where in the first part structure of Bayesian network will be described and in the second part we will describe multi agent system implemented for sampling data for conditional probabilities tables (CPT) construction. I the last part of paper we will describe implementation of Bayesian proprioceptor and show experimental results.

2 Observer Network and Data Samples

Observer network is a network of observer nodes each of them used for forest fire monitoring of the surrounding area [4]. A single observer node consists of following parts:

-video camera with pan/tilt/zoom capability so it can be used for monitoring 360 degrees surrounding area,
-a mini meteorological station for environmental parameters monitoring,
-sensors for processing parameters monitoring (ie. temperature of the equipment, voltage of solar powered battery)
-embedded data server for data preprocessing and
-communication equipment.

A typical observer node, deployed on top of Marjan hill in Split is shown in figure 1.

Forest fire detection is performed in cycles with following steps:

1. Data collection
2. Syntactical data validation
3. Semantical data validation
4. Fire detection
5. Alarm post processing

The first three steps are task of Network observer, and steps 4 and 5 are task of phenomenon observer. Collection of agents for both network and phenomenon observer are together forming observer network multi agent system. Data collection is performed through communication of CollectorAgent and MasterAgents. A CollectorAgent is responsible for collecting data of one observer node. Cyclically (typically every 10 mintes) the CollectorAgent asks each MasterAgent of the observer node its values. The values are then grouped into data sample or information vector, which is a vector of values, each representing one sensed or virtually sensed data. Each value is checked for syntactical error, and then information vector is checked for semantic error. Each data is separately stored in database for future presentation or analysis. A special kind if CollectorAgent is used for video camera image collection, and images are stored in image base, and analysed on several levels so they can provide additional data (such as level of brightness).

Fig. 1. Observer Node

In this paper our focus will be on network observer, especially on semantical data validation. A Bayesian network for semantical data validation will be designed and implemented. Semantical data validation is important in false alarm reduction because semantically invalid data has high probability for causing false alarm.

3 Bayesian Network Design

Occurrence of false alarm in forest fire perception system relies on many parameters which are mutually condicionaly dependant. Bayesian network concept is appliable for probabilistic inference about forest fire [9].

Bayesian network consists of nodes, relationship among nodes and conditional probability tables assigned for each node. Construction of Bayesian network is combination of implicit definition of network structure and empirical construction of conditional probabilities tables using past data. Multi agent Bayesian network implementation has also been recognized by other authors [1,2]

3.1 Bayesian Network Structure Design

Construction of Bayesian network starts with identification of concepts that will become variables in Bayesian network. Each of those concepts corresponds to a particle of data that will later be used in deduction of forest fire alarm. Individual particles of data identified in one observer node are shown in figure 2.

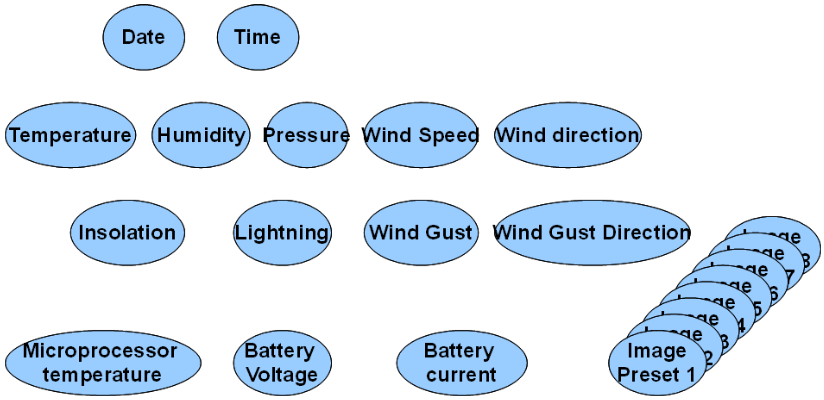


Fig. 2. Distinguished data that can be used in reasoning

Each of those concepts hold values that are typically continuous. Using continuous values is rather impractical, so discretisation of values based on histogram was performed. For each of the values we have visually determine the range of possible values from histogram that are shown in figure 3 and then split that range into more ranges for discrete values and ranges with lower histogram in fewer values We have identified up to 7 discrete values each of a value of centroid representing a range of vales for each variable. It is important to emphasise that the implementation of MAS is such that these ranges and its centroids are easily changed without changes in implementation.

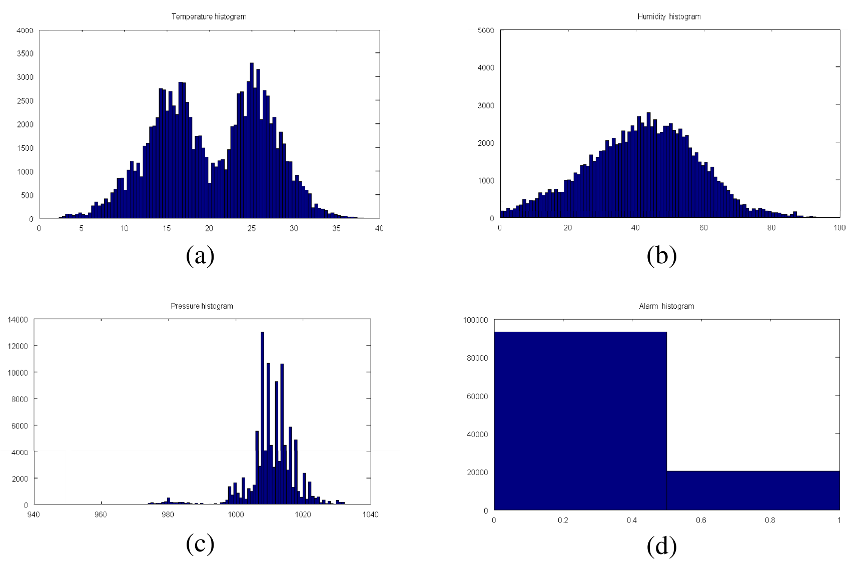


Fig. 3. Histogram of (a) Temperature, (b) Humidity, (c) Pressure, (d) Alarm for data collected during year of 2011

The descrete ranges are formally described by boundary values. For example, if we have 6 ranges : <-∞,0>,<0,5>,<5,10>,<10,15>,<15,20> …. we can describe them with only margin values 0, 5, 10, 15, 20 ... We have used the ranges described as below, but ranges are easily adapted if it is necessary to have more precise values:

Ranges(Temp)=0,5,10,15,20,25,30,35
Ranges(Hum)=20,40,60,80
Ranges(Press)=900,980,990,1000,1010,1020
Ranges(Alarm)=0.5
Ranges(Fire)=0.5

Second step was identification of connections between concepts. In this step a trade-off was made on behalf of number of connections , because increasing the number of connection, the system would be more accurate, but the complexity of the system increases dramatically with every new connection, so only necessary connections are used.

The result is Bayesian network structure, as shown in figure 4.

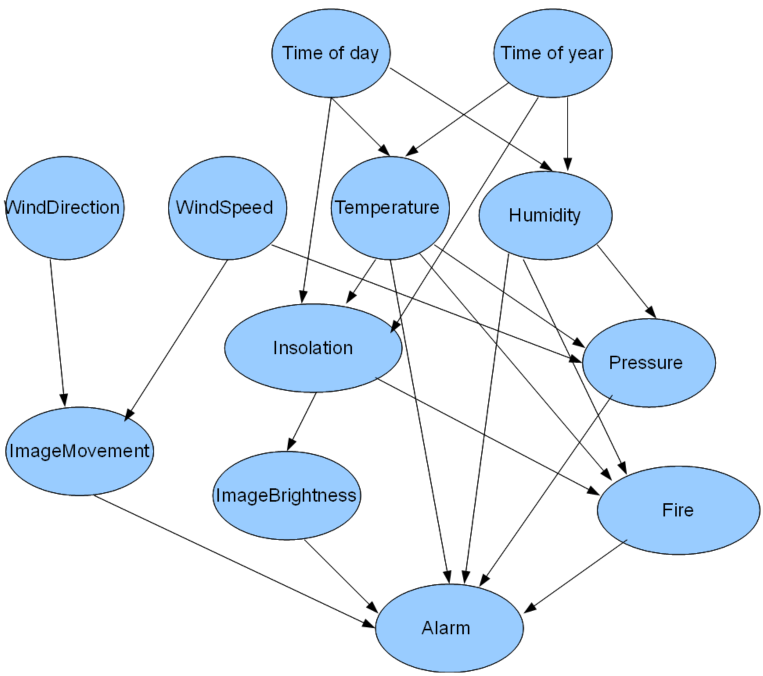


Fig. 4. Bayesian network for forest fire alarm

Identifying the Bayesian network as a directed acyclic graph is only half of the job. Conditional probabilities tables must also be filled in with data so Bayesian network can be used in reasoning.

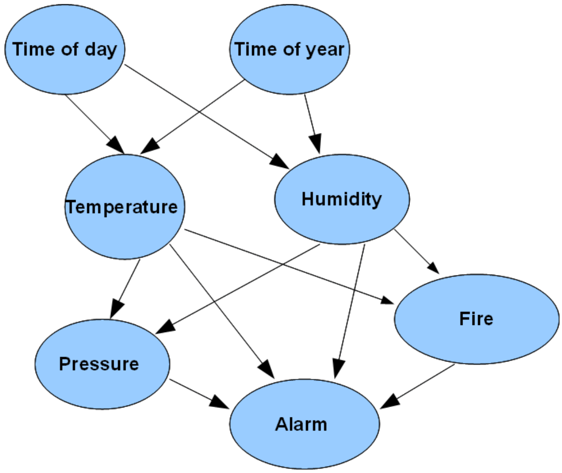


Fig. 5. Simplified Bayesian network for Observer Network

To simplify the description we will use simplified version of Bayesian network shown in figure 5.

3.2 CPT Construction

A multi agent system shell was implemented with task of conditional probability tables (CPT) construction. The multi agent shell consists of implementation of agent types with general behaviours, but during run time agents are instantiated as needed. Agents forming MAS shell for sampling are shown in figure 5.

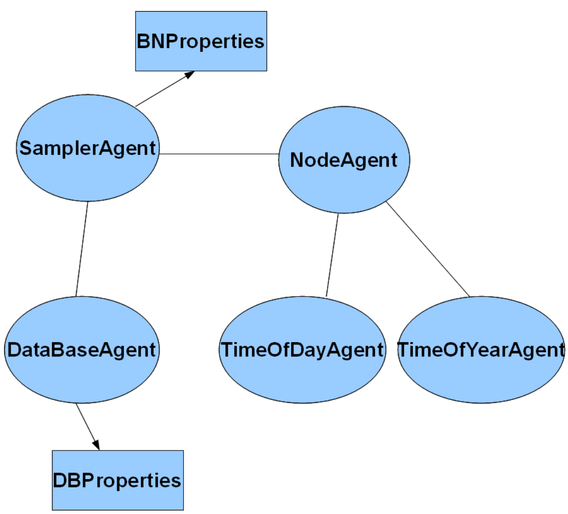


Fig. 6. Agents of multi agent shall for CPT construction

The SamplerAgent forms cyclically a sample of data retrieved on the same collection cycles and stored in database. A sample is then used for adding information to CPT. The SamplerAgent sends each NodeAgent necessary value, while NodeAgent is responsible only for its own table construction.

4 Reasoning with Bayesian Network

Once constructed and filled with CPT tables Bayesian network can be used as a model of environment. The model can be mostly used as proprioceptor classifier. A measure, or weight of soundness (WOS) can be attached to each new sample of collected values, so the less the measure is, the data is more probable to be invalid. Especially, it a sample contains alarm and whole sample has low weight of soundness, the alarm has high probability measure of being false alarm.

Another use of Bayesian network reasoning is approximating missing values with most probable value taking into account surrounding values in data sample.

4.1 Proprioceptor Classifier

Proprioceptor is a part of observer nework used for sensor network self perception, and its task is detection of soundness of data collected from sensor network. Proprioception is performed in two steps: syntactical validation and semantical validation.

Syntactical validation is simpler step and is performed by checking if data is from the range of allowed values for the specific kind as shown in Figure 7.

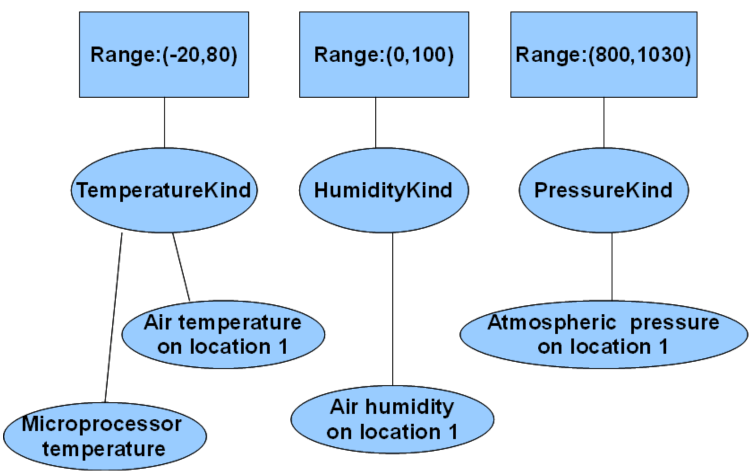


Fig. 7. Syntactical validation of data using allowed ranges of values

It will be shown how Bayesian network constructed in previous chapters can be used for semantically data validation and thus false alarm reduction.

Bayesian network with probabilities tables can be used as a model of the environment, having connections among concepts identified and quantified with conditional probabilities distributions.

Reasoning is performed on a sample of values collected from the sensor network. Top-level concepts, that is nodes without parents, are nodes holding values that are independent on sensor network, time of day and time of year. We will assume that computer clock is well tuned and in each sample, we have actual correct time end date, so we will hold these values fixed. This will reduce the number of rows in CPT having TOD and TOY values as conditions to those fixed values. Now, CPT of nodes depending only on these values is reduced only to probability distribution of its own value. Comparing the centroid to which the value from the sample belongs with probability distribution we can weight the likelihood of data coming from the sensors, and so quantify soundness of data.

A cumulative weighed factor obtained by multiplying weights of each nodes can be used as a classifier for soundness of data, that is the semantical validation, weighting the sense of data and choosing data that are suspicious to belong to the environment the Bayesian network models.

If we denote sample of data with {S} and nodes of Bayesian network with {N} we can describe the Bayesian network

S={Datetime,T,H,P,A,F}
N={TOD,TOY,Temp,Hum,Press,Fire,Alarm}

Structure of Bayesian network can be described by stating dependencies of each node, where first dependency is value from the sample the node value depends on, and rest of the dependencies are conditional dependency nodes.

Dependencies(TOD)=Datetime
Dependencies(TOY)=Datetime
Dependencies(Temp)=T;TOD,TOY
Dependencies(Hum)=H;TOD,TOY
Dependencies(Press)=P;Temp,Hum
Dependencies(Fire)=F;Temp,Hum
Dependencies(Alarm)=A;Temp,Hum,Press,Fire

The weight of soundness is calculated by multiplying conditional probability values in rows of CPT the sample belongs to. The formal equation of WOS is given by equation

$$WOS = \prod_{Ni} P(V(N_i) = V_i \mid \{V_j = V(N_j); \forall Dependencies(N_i)\}) \quad (1)$$

5 Deployment and Results

BayesProprioceptor is deployed as a part of observer network multi agent shell. Each sample of data collected from the sensor network is forwarded to Bayesian Proprioceptor clasiffier and whole sample is assigned measure of soundness.

Also, probability of false alarm is calculated so raising alarm in situation with low false alarm probability more likely correspond to true alarm.

A snapshot of several samples with assigned measure of soundness is shown in table 1. Sample of data is collected in 2012, while CPT are constructed using data collected in 2011.

Table 1. Resulting Weight of Soundness for several samples of data

S						
Datetime	***T***	***H***	***P***	***F***	***A***	***WOS***
1326195902016	10.9	27.5	1013.9	0	0	0.03318825
1326195921599	10.3	26.8	1013.9	0	1	0.015184167
1325304416600	8.7	20.7	997.6	0	1	7.7006 E-6
1325217860807	9.9	55.2	1005.7	0	1	2.6336 E-4

6 Conclusion

In this paper we have described implementation of sampling MAS which is general and can be used for Bayesian network sampling because of its modular implementation. We have described a classifier for sensor network data validation based on Bayesian network CPT values.

As advantages of such approach we will say that probabilistic reasoning is similar to human reasoning, once CPT are created it is computationally cheep and fast. It is also possible to automate learning about environment with periodically adapted tables.

The proposed system has also several disadvantages. Large amount of data must be collected before CPT are created, so it can not be used for newly deployed system.

Naïve Bayes assumption about conditional dependencies of two contiguous data was made, which is not correct. More accurate model would include conditional probability of previous value, so our future work will be directed in implementation of Markov chain proprioceptor classifier.

References

1. Arrue, B.C., Ollero, A., de Dios, J.R.M.: An intelligent system for false alarm reduction in infrared forest fire detection. IEEE Intelligent Systems & Their Applications 15(3), 64–73 (2000)
2. Pavlin, G., de Oude, P., Maris, M., Nunnink, J.: Thomas Hood: A multi-agent systems approach to distributed bayesian information fusion. Information Fusion 11(3), 267–282 (2010)
3. Xiang, Y.: Distributed Multi-Agent Probabilistic Reasoning With Bayesian Networks. In: Raś, Z.W., Zemankova, M. (eds.) ISMIS 1994. LNCS, vol. 869, pp. 285–294. Springer, Heidelberg (1994)
4. Šerić, L., Stipaničev, D., Štula, M.: Observer Network and Forest Fire Detection. Information Fusion 12(3), 160–175 (2011)

5. Heckerman, D., Mamdani, A., Wellman, M.P.: Real-world applications of Bayesian networks. Commun. ACM 38(3), 24–26 (1995)
6. Baumgartner, K., Ferrari, S., Palermo, G.: Constructing Bayesian networks for criminal profiling from limited data. Know.-Based Syst. 21(7), 563–572 (2008)
7. Sahami, M., Dumais, S., Heckerman, D., Horvitz, E.: A Bayesian approach to filtering junk e-mail. In: AAAI 1998 Workshop on Learning for Text Categorization Papers from the 1998 Workshop, vol. 62, 98–05 (1998)
8. Wagner, S.: A Bayesian network approach to assess and predict software quality using activity-based quality models. In: Proceedings of the 5th International Conference on Predictor Models in Software Engineering (PROMISE 2009). ACM, New York (2009)
9. Dlamini, W.M.: Application of Bayesian networks for fire risk mapping using GIS and remote sensing data. GeoJournal 76(3), 283–296 (2011)

Supervision of Agents Modelling Evacuation at Crisis Situations

František Čapkovič*

Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovak Republic
Frantisek.Capkovic@savba.sk
http://www.ui.sav.sk/home/capkovic/capkhome.htm

Abstract. The main aim of this paper is to point out that the agent cooperation based on supervision can be utilized also in crisis situations. The approach is applied to an evacuation of an endangered area (EA). The supervisors for agents modelling crisis situations are synthesized and presented here. This makes possible not only the agent cooperation in itself but also to check the evacuation process and guarantee full evacuation of people from EA. Place/transitions Petri Nets (P/T PN) are utilized in order to model the elementary autonomous agents being modules of EA. The supervision of agents is based on the DES (discrete-event systems) control theory. It is realized by means of known DES supervision methods. The proposed approach is illustrated in details on the supervision-based cooperation of agents in a case study of evacuation. The corresponding evacuation workflow is proposed by means of PN too. Moreover, the supervision is applied also in case of the workflow.

Keywords: Agent, cooperation, crisis situation, discrete-event systems, evacuation, modularity, place/transition Petri nets, supervision, workflow.

1 Introduction and Preliminaries

To manage the evacuation process from EA in a crisis situation (particularly in case of a panic) flexible strategies for safety solving the problem are required. Especially, it is necessary to find safety and free escape routes [10]. Of course, immediate information from the real system (the real scenario) is valuable on this way. A network of distributed sensors yields necessary information. However, a model representing the layout (schema) of EA and main aspects of the evacuation dynamics has also an importance at synthesizing a suitable supervisor. The model yields a possibility to describe and analyze the sequences of steps at the escape from the EA. It can be built from elementary modules (agents) describing and modelling EA structure and evacuation dynamics. By the term agent a material entity (elementary module) able to cooperate (at least with a

* Partially supported by the Slovak Grant Agency for Science VEGA under grant # 2/0075/09.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 24–33, 2012.

supervisor) is deemed here. In the case study introduced below, agents are represented by doorways equipped by sensors. Because the model has character of DES, it can be modelled by P/T PN. There exist methods in DES control theory how to synthesize the supervisors [7]. Supervisors will ensure an intended agent cooperation. On the other hand the models can be useful also at prevention crisis situations in some areas (e.g. buildings) already at their design. Namely, they allow to analyze buildings as to the evacuation in case of accidents in advance (i.e. already at their design). As to workflow modelling there are approaches using PN - see e.g. [12, 14, 13]. We will use P/T PN for modelling EA structure and evacuation dynamics (here the PN-based workflow modelling will be utilized too) as well as for the supervisor synthesis. Because existing PN tools do not allow the supervisor synthesis, computing by MATLAB will be performed.

The P/T PN [11] are used here in the process of modelling the agents, agent cooperation and supervision. The earlier results concerning the agents cooperation [1, 4, 5, 2, 3, 6] as well as the theory of supervision [7–9] can be utilized. As to the structure P/T PN are bipartite directed graphs $< P, T, F, G >$ with P, T, F, G being, respectively, the set of places p_i, $i = 1, \ldots, n$, the set of transitions t_j, $j = 1, \ldots, m$, the set of directed arcs from places to transitions and the set of directed arcs from transitions to places. Here, $P \cap T = \emptyset$, $F \cap G = \emptyset$. Moreover, P/T PN have their dynamics $< X, U, \delta, \mathbf{x}_0 >$ with $X, U, \delta, \mathbf{x}_0$ being, respectively, the set of state vectors (i.e. vectors of marking the places), the set of vectors of discrete events (state vectors of transitions), the transition function and the initial state vector. Here, $X \cap U = \emptyset$, $\delta : X \times U \rightarrow X$. The formal expression of δ can be rewritten into the system form as follows

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}.\mathbf{u}_k,\ \mathbf{B} = \mathbf{G}^T - \mathbf{F},\ k = 0, ..., N \tag{1}$$

$$\mathbf{F}.\mathbf{u}_k \leq \mathbf{x}_k \tag{2}$$

where k is the discrete step of the dynamics development; $\mathbf{x}_k = (\sigma^k_{p_1}, ..., \sigma^k_{p_n})^T$ is the n-dimensional state vector; $\sigma^k_{p_i} \in \{0, 1, ..., c_{p_i}\}$, $i = 1, ..., n$ express the states of atomic activities by 0 (passivity) or by $0 < \sigma_{p_i} \leq c_{p_i}$ (activity); c_{p_i} is the capacity of p_i; $\mathbf{u}_k = (\gamma^k_{t_1}, ..., \gamma^k_{t_m})^T$ is the m-dimensional control vector; its components $\gamma^k_{t_j} \in \{0, 1\}$, $j = 1, ..., m$ represent occurring of elementary discrete events (e.g. starting or ending the activities, failures, etc.) by 1 (presence of the corresponding discrete event) or by 0 (absence of the event); $\mathbf{B}, \mathbf{F}, \mathbf{G}$ are matrices of integers; $\mathbf{F} = \{f_{ij}\}_{n \times m}$, $f_{ij} \in \{0, M_{f_{ij}}\}$, expresses the causal relations among the states (as causes) and the discrete events occurring during the DES (discrete-event systems) operation (as consequences) by 0 (nonexistence of the relation) or by $M_{f_{ij}} > 0$ (existence and multiplicity of the relation); $\mathbf{G} = \{g_{ij}\}_{m \times n}$, $g_{ij} \in \{0, M_{g_{ij}}\}$, expresses analogically the causal relations among the discrete events (as causes) and the DES states (as consequences); $(.)^T$ symbolizes the matrix or vector transposition.

Just such an exact mathematical expression of P/T PN yields, in contrast to high-level PN, the possibility to deal with the PN models in analytical terms.

Modular Structure of PN-Based Models of Agents. Having modules - the PN models of particular agents - we can think about building a suitable global structure from such modules. In case of N_A autonomous agents the structural matrices of the global model can have the following form

$$\mathbf{F} = \big(\mathrm{diag}(\mathbf{F}_i)\ \ \mathbf{F}_c\big)\,;\ \mathbf{G}^T = \big(\mathrm{diag}(\mathbf{G}_i^T)\ \ \mathbf{G}_c^T\big)$$

where $\mathbf{F}_i$, $\mathbf{G}_i$, $i = 1, \dots, N_A$, represent the parameters of the PN model of the agent A_i; $\mathbf{F}_c = (\mathbf{F}_{c_1}^T, \mathbf{F}_{c_2}^T, ..., \mathbf{F}_{c_{N_A}}^T)^T$, $\mathbf{G}_c^T = (\mathbf{G}_{c_1}, \mathbf{G}_{c_2}, ..., \mathbf{G}_{c_{N_A}})^T$ represent the structure of the interface between the cooperating agents. More details about other kinds of modular structures can be found in [6]. The modularity can be comprehended in a wider sense. Namely, the model of a large-scale EA can be built from the models of local EAs (built from elementary agents). Several levels of such an embedding can be used.

Supervision and the Supervisor Synthesis. The classical feedback control typical for continuous systems cannot be applied to DES. However, some kinds of supervision used in the DES control theory can be applied in order to affect the system and to force it a desired behaviour, namely (in our case), to compel it to a purposeful cooperation of subsystems (modules/agents). In P/T PN models of DES the supervisors can be synthesized in analytical terms.

Here, two kinds of supervision are utilized, namely: (i) the supervision based on P-invariants of P/T PN; (ii) the extended supervision based on the state vectors $\mathbf{x}$, control vectors $\mathbf{u}$ and Parikh's vector $\mathbf{v}$. The former approach utilizes definition of P-invariant as $\mathbf{w}^T.\mathbf{B} = \mathbf{0}$ or in case of more P-invariants as $\mathbf{W}^T.\mathbf{B} = \mathbf{0}$, where $\mathbf{W}$ is the matrix with rows being P-invariants. This approach operates only with the state vector $\mathbf{x}$. The latter approach is typical by usage of the state vector $\mathbf{x}$, the control vector $\mathbf{u}$ and, finally, the Parikh's vector $\mathbf{v}$ which is defined by developing the system (1) as follows: $\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{B}.\mathbf{u}_0$; $\mathbf{x}_2 = \mathbf{x}_1 + \mathbf{B}.\mathbf{u}_1 = \mathbf{x}_0 + \mathbf{B}.(\mathbf{u}_0 + \mathbf{u}_1)$ etc. Hence,

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{B}.(\mathbf{u}_0 + \mathbf{u}_1 + \cdots + \mathbf{u}_{k-1}) = \mathbf{x}_0 + \mathbf{B}.\mathbf{v} \tag{3}$$

Just the vector $\mathbf{v} = \sum_{i=0}^{k-1} \mathbf{u}_i$ is the Parikh's vector. It gives us information about how many times the particular transitions are fired during the development of the system dynamics from the initial state $\mathbf{x}_0$ to the final state $\mathbf{x}_k$.

(i) Supervisor Synthesis Based on P-invariants: Here, a set of n_s inequalities in the vector form $\mathbf{L}_p.\mathbf{x} \leq \mathbf{b}$ expressing conditions for a desired behaviour of the system is prescribed. Then, the inequalities are transformed into equations by introducing the auxiliary variables (slacks). Consequently, the definition of P-invariants is replaced by

$$(\mathbf{L}_p, \mathbf{I}_s).\begin{pmatrix}\mathbf{B}\\ \mathbf{B}_s\end{pmatrix} = \mathbf{0};\quad \mathbf{L}_p.\mathbf{B} + \mathbf{B}_s = \mathbf{0};\quad \mathbf{B}_s = -\mathbf{L}_p.\mathbf{B};\quad \mathbf{B}_s = \mathbf{G}_s^T - \mathbf{F}_s \tag{4}$$

$$(\mathbf{L}_p\,|\,\mathbf{I}_s).\begin{pmatrix}\mathbf{x}_0\\ {}^s\mathbf{x}_0\end{pmatrix} = \mathbf{b}\,;\quad {}^s\mathbf{x}_0 = \mathbf{b} - \mathbf{L}_p.\mathbf{x}_0$$

In such a way the structure of the supervisor and its initial state were obtained. They make possible to create the augmented (i.e. supervised) system

$$\mathbf{x}_a = \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \end{pmatrix}; \quad \mathbf{F}_a = \begin{pmatrix} \mathbf{F} \\ \mathbf{F}_s \end{pmatrix}; \quad \mathbf{G}_a^T = \begin{pmatrix} \mathbf{G}^T \\ \mathbf{G}_s^T \end{pmatrix} \tag{5}$$

(ii) Supervisor Synthesis Based on the Extended Method: Here, the general linear constraints expressing the conditions are extended [9] into the form as follows

$$\mathbf{L}_p.\mathbf{x} + \mathbf{L}_t.\mathbf{u} + \mathbf{L}_v.\mathbf{v} \leq \mathbf{b} \tag{6}$$

where $\mathbf{L}_p$, $\mathbf{L}_t$, $\mathbf{L}_v$ are, respectively, $(n_s \times n)-$, $(n_s \times m)-$, $(n_s \times m)-$dimensional matrices. When $\mathbf{b} - \mathbf{L}_p.\mathbf{x} \geq \mathbf{0}$ is valid - see e.g. [9] - the supervisor with the following structure and initial state

$$\mathbf{F}_s = \max(\mathbf{0},\, \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v,\, \mathbf{L}_t) \tag{7}$$

$$\mathbf{G}_s^T = \max(\mathbf{0},\, \mathbf{L}_t - \max(\mathbf{0},\, \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v)) - \min(\mathbf{0},\, \mathbf{L}_p.\mathbf{B} + \mathbf{L}_v) \tag{8}$$

$${}^s\mathbf{x}_0 = \mathbf{b} - \mathbf{L}_p.\mathbf{x}_0 - \mathbf{L}_v.\mathbf{v}_0 \tag{9}$$

guarrantees that constraints are verified for the states resulting from the initial state. Here, the max(.) is the maximum operator for matrices. However, the maximum is taken element by element. Namely, in general, for the matrices $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$ of the same dimensionality $(n \times m)$, the relation $\mathbf{Z} = \max(\mathbf{X}, \mathbf{Y})$ it holds that $z_{ij} = \max(x_{ij}, y_{ij})$, $i = 1, ..., n$, $j = 1, ..., m$.

2 Case Study - Evacuation of Persons from the Building

Consider the EA given on the left in Fig. 1. It consists of two bands of rooms and the corridor among the bands. Such an architecture is typical e.g. for hotels, colleges, etc. For simplicity, consider only two rooms in any band. Then, there are four rooms R1 - R4 and the corridor R5 in EA. The corridor is accessible from any room. R2 and R4 have, respectively, own emergency exits E2 and E3 why R5 has the main emergency exit E1. The escape routs depends on the doors. While doors D1, D3 and the emergency exits E1 - E3 are one-way (i.e. suitable only for the escape in the direction outside from the room), the doors D2 - D4 are two-way (i.e. suitable for the escape in the direction outside from the room and vice-versa - e.g. in case when E1 does not operate or when it is crowded). The EA may also be a flat, a segment of a hotel floor, but it can also be an arbitrary kind of EA with different shapes of rooms and escape routs. In order to model EA, it is necessary to distinguish two kinds of PN-models of doors [10], namely, the one-way door PN-model and two-way door one. The differences are evident not only from the schema of EA given in the left part of Fig. 1 but also from the PN-models of both kinds of doors given in the right part of Fig. 1. Namely, the P/T PN model of the one-way door is given in the center

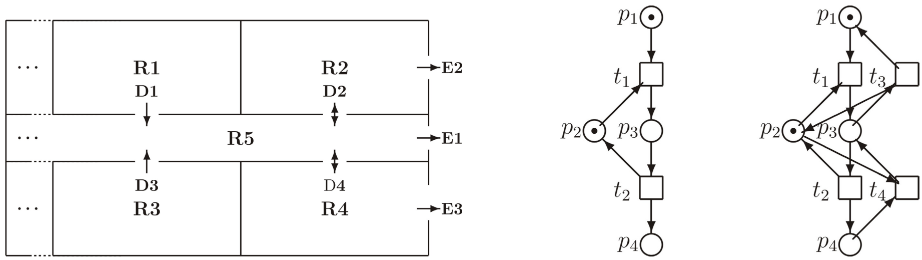


Fig. 1. On the left side is the schema of the EA. On the right side are the P/T PN-based models of the one-way door (on the left) and that of two-way door (on the right).

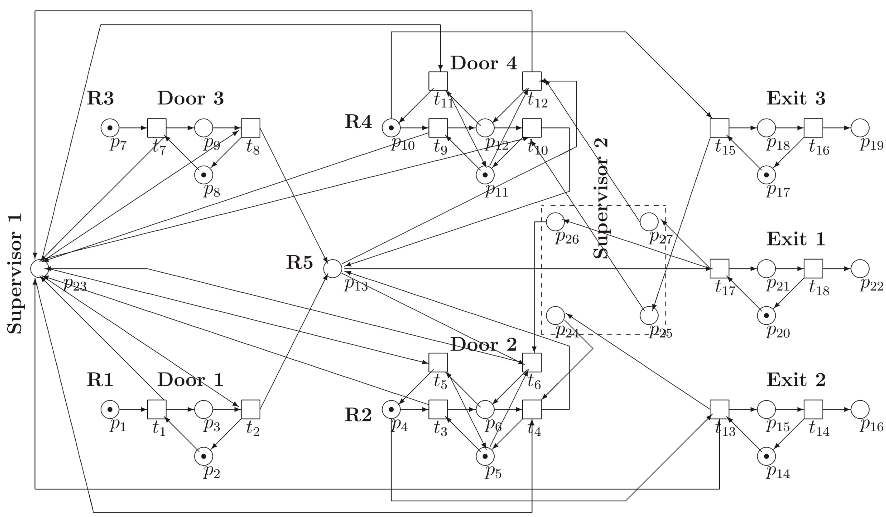


Fig. 2. The P/T PN-based model of the EA

of Fig. 1, while the P/T PN model of the two-way door is given on the right. The place p_1 models a room from which the door exits while the place p_4 models the room which the door enters. The two-way doors can be passed from both sides. The place p_2 represents the availability of the door. Of course, the door can be passed only in case when it is available. Finally, the place p_3 represents the process of passing the door. In other words, the mean of the PN places is the following: p_1 - represents a room and the token inside expresses a state of the room - i.e a presence of a person in this room; p_2 - models availability of the door - a token inside means that the door is available, while in the opposite case the door is not available; p_3 - models passing the door - when it contains a token it means that e.g. a person just passes this door; p_4 - models an external room or the corridor. Firing the transitions t_1 - t_4 (if enabled) makes possible to control marking dynamics. The incidence matrices of the PN models of agents, i.e. $\mathbf{F}_1$, $\mathbf{G}_1^T$ for the one-way door and $\mathbf{F}_2$, $\mathbf{G}_2^T$ for the two-way door are the following

$$\mathbf{F} = \begin{pmatrix} 1\,0 \\ 1\,0 \\ 0\,1 \\ 0\,0 \end{pmatrix} ; \; \mathbf{G}^T = \begin{pmatrix} 0\,0 \\ 0\,1 \\ 1\,0 \\ 0\,1 \end{pmatrix} ; \; \mathbf{F} = \begin{pmatrix} 1\,0\,0\,0 \\ 1\,0\,0\,1 \\ 0\,1\,1\,0 \\ 0\,0\,0\,1 \end{pmatrix} ; \; \mathbf{G}^T = \begin{pmatrix} 0\,0\,1\,0 \\ 0\,1\,1\,0 \\ 1\,0\,0\,1 \\ 0\,1\,0\,0 \end{pmatrix}$$

Here, EA consists of agents being two kinds of doors. In general, EA in itself can be understood to be an agent (a complex door) with paths inputting from and outputting to complex rooms connecting this EA with other such agents.

2.1 PN-Based Model of the Endangered Area

Now, we can create the model of the EA by means of modularity approach. The PN-based model of EA can be built by choosing the corresponding rooms of EA to be the input/output rooms of the doorways. It is necessary to take into account that R5 is external room for all of the doors D1, D2, D3, D4. Thus, we obtain the PN model given in Fig. 2. Simultaneously, we could see that the one-way doors from R1 and R3 enter the corridor as well as the two-way doors from R2 and R4. Moreover, R2 and R4 enter, respectively, the one-way doors (exits) E2 and E3. The corridor R5 enter the one-way door (exit) E1. Consider the PN-based models of doorways (being modules of the PN model of EA) to be, generally spoken, agents. Name them "material" agents, in spite of the fact that the term agents is usually used almost exclusively for software agents. However, at this moment we have in mind only the structure given in Fig. 2 without the Supervisor 1 (S1) and the Supervisor 2 (S2). Namely, both supervisors will only be synthesized thereinafter. As to the structure the PN model corresponds to the real EA. Dynamics of the evacuation process is described with the mathematical model (1) where the structure of the incidence matrices $\mathbf{F}$, $\mathbf{G}^T$ depends on agents interactions. The animation of the tokens (typical for PN graphical tools) will be replaced by step-by-step computing the state vectors in the simulation tool MATLAB, because no PN tool is able to synthesize the supervisors. More details how to do can be found in [1], [4]. When no supervisors are used the adjacency matrix of the PN model reachability tree has a large dimensionality. There are many possible states in the evacuation process, namely, $N = 1849$ different states. The reason for this is that the movement of the system is not coordinated. The autonomous agents behave completely free. They are not organized so far, i.e. no interferences are performed yet. Such a model comprehends all possible state trajectories including all of escape roads from EA. Consequently, in case of a panic the situation cannot be managed without an entity organizing the escape. Therefore, a way how to deal with such a problem is to supervise the process by the supervisor(s). Now, the supervisors S1, S2 will be synthesized here. The synthesis of S1 will start from the EA initial state

$$\mathbf{x}_0 = \left(1\,1\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,0\,0\right)^T .$$

Their structure and marking were already displayed in Fig. 2 in order to avoid drawing such a big figure two times.

(i) Synthesizing the supervizor S1: Taking into account the condition $p_1 + p_4 + p_7 + p_{10} + p_{13} \leq b$, where $b = 4$ means the global number of tokens (e.g. persons) in the rooms including the corridor we obtain

$$\mathbf{L}_p = \begin{pmatrix} 1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0 \end{pmatrix};\ \mathbf{b} = \begin{pmatrix} 4 \end{pmatrix}$$

$$\mathbf{F}_s = \begin{pmatrix} 01011001011000000 \end{pmatrix};\mathbf{G}_s^T = \begin{pmatrix} 101001101001101010 \end{pmatrix};\mathbf{x}_{s0} = \begin{pmatrix} 0 \end{pmatrix}$$

The number of states of the system supervised only by the S1 is $N = 1849$ too. However, the supervisor checks the number of evacuated persons. It is given by the place p_{23} together with its connections with other parts of the model.

(ii) Synthesizing the supervizor S2: S2 will be synthesized from the initial state

$$\mathbf{x}_{a0} = (\mathbf{x}_0^T\ \mathbf{x}_{s0}^T)^T = \begin{pmatrix} 1\,1\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,0\,0\,|\,0 \end{pmatrix}^T$$

In general, it can happen that during the evacuation process some doors can have a higher priority than others. In our case, the evacuation of the rooms R2, R4 should prefer the exits E2, E3 before doors D2, D4, respectively. Thus, the exit E1 is able to manage the evacuation of the rooms R1 and R3 better. At solving priority problems the Parikh's vector (3) is very useful. The conditions described verbally can be expressed in mathematical terms as follows $v_4 \leq v_{13}$; $v_{10} \leq v_{15}$; $v_6 \leq v_{17}$; $v_{12} \leq v_{17}$. Here, v_i are the components of the Parikh's vector and they concern the PN transitions with the same indices. Consequently,

$$\mathbf{L}_v = \begin{pmatrix} 0&0&0&1&0&0&0&0&0&0&0&0&-1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0&0&0&-1&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&-1&0 \end{pmatrix};\ {}^2\mathbf{b} = \begin{pmatrix} 0\\0\\0\\0 \end{pmatrix}^T;\ {}^2\mathbf{x}_{s0} = \begin{pmatrix} 0\\0\\0\\0 \end{pmatrix}^T$$

$${}^2\mathbf{F}_s = \begin{pmatrix} 000100000000000000 \\ 000000000100000000 \\ 000001000000000000 \\ 000000000001000000 \end{pmatrix};{}^2\mathbf{G}_s^T = \begin{pmatrix} 000000000000100000 \\ 000000000000001000 \\ 000000000000000010 \\ 000000000000000010 \end{pmatrix}$$

The supervisor S2 is given by places p_{24}, p_{25}, p_{26}, p_{27} (included in Fig. 2 into the dashed box) together with their connections with other parts of the PN model. S1 and S2 were already introduced in Fig. 2 in order to save space and not to repeat such a big figure once more. The number of states of the system supervised by both the S1 and the S2 is $N = 608$, i.e. less than 1/3 from 1849.

2.2 PN-Based Modelling the Workflow of the Evacuation Process

Usually, it is not necessary to compute the complete state space. Perhaps only at the design of buildings it is useful from the escape safety point of view. Therefore, let us create now the PN-based model of the wokflow [12, 14, 13] of the evacuation process described above. Its simplest form is given on the left in Fig. 3. Here the sense of the places p_i and transitions t_j is completely different like before, of course. Namely, p_1 expresses the start of the evacuation process; p_2 - p_6 represent the rooms R1 - R5 to be evacuated; p_7, p_8, p_9 express, respectively, the exits E2, E3, E1; p_{10} represents the checking point (it finds if all of the rooms were already evacuated); p_{11} expresses the end of evacuation process. Here,

$$\mathbf{F} = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&1&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&5 \\ 0&0&0&0&0&0&0&0&0&0&0&0 \end{pmatrix}; \mathbf{G}^T = \begin{pmatrix} 0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&0&0 \\ 1&1&0&1&0&0&1&1&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&1&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1 \end{pmatrix}; \mathbf{x}_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

The the state space of such a system has $N = 327$ states. Also here we can apply

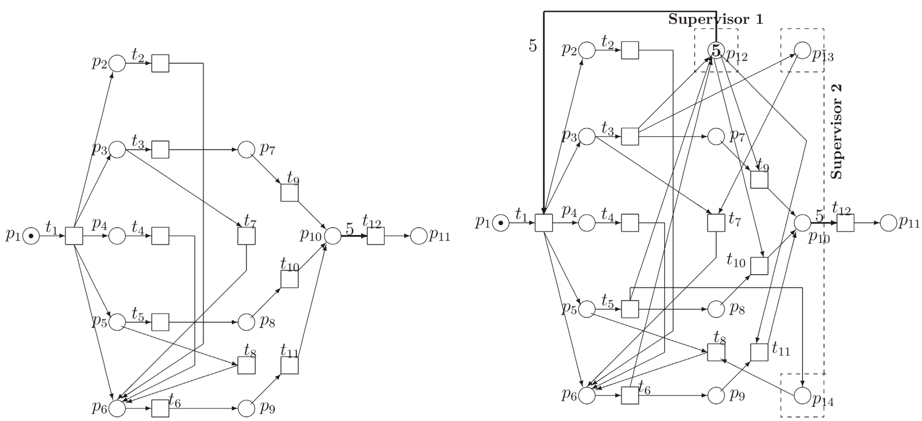


Fig. 3. On the left is the P/T PN-based workflow of the evacuation process from EA while on the right is the supervized P/T PN-based workflow

the supervision like in the previous approach, in order to improve the quality of the evacuation process and reduce the state space. For the S1 the condition $p_2 + p_3 + p_4 + p_5 + p_6 + p_{10} \leq 5$ has to be met, because there are 5 rooms to be evacuated and a checker. For the S2 the conditions $v_7 \leq v_3$; $v_8 \leq v_5$ has to be satisfied. Then we obtain the supervised workflow given on the right in Fig. 3. Thus, for the synthesis of S1 we have

$$\mathbf{x}_0 = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0 \end{pmatrix}; \; \mathbf{L}_p = \begin{pmatrix} 0&1&1&1&1&1&0&0&0&1&0 \end{pmatrix}; \; \mathbf{b} = \begin{pmatrix} 5 \end{pmatrix}$$

$$\mathbf{F}_s = \begin{pmatrix} 5&0&0&0&0&0&0&0&1&1&1&0 \end{pmatrix}; \; \mathbf{G}_s^T = \begin{pmatrix} 0&0&1&0&1&1&0&0&0&0&0&0 \end{pmatrix}; \; \mathbf{x}_{s0} = \begin{pmatrix} 5 \end{pmatrix}$$

The the state space of the system only with the S1 has $N = 327$ states. However, we can add S2 to S1. At its synthesis for the augmented system (original system + S1) we have

$$^2\mathbf{x}_0 = \begin{pmatrix} 1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \mid 5 \end{pmatrix}^T$$

$$\mathbf{L}_v = \begin{pmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix};\ ^2\mathbf{b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix};\ ^2\mathbf{x}_{s0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$^2\mathbf{F}_s = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0 \end{pmatrix};\ ^2\mathbf{G}_s^T = \begin{pmatrix} 0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0 \end{pmatrix}$$

The the state space of the system with both the S1 and the S2 has $N = 227$ states, i.e. less than 327 about for 1/3.

3 Conclusion

Two kinds of PN-based approaches were presented here. First, P/T PN were utilized (i) for modelling structure of EA by a modular approach, where the PN modules being "material" agents were defined. In the introduced case study PN-based models of two kinds of doorways were understood to be such agents; (ii) for modelling dynamics of the evacuation process at occurring a crisis situation; (iii) for synthesizing the supervisors P/T PN ensuring desired form of evacuation. Next, the PN-based modelling the workflow of the evacuation process. Moreover, a trial with the supervised workflow process was tested in this case. In both approaches two supervisors S1 and S2 synthesized by means of mutually different methods were used. Namely, the S1 in both cases was synthesized by means of P/T PN P-invariants while the S2 was synthesized by means of extended method based on the state vector, control vector and Parikh's vector. The advantage of the former approach (based on the P/T PN model) is the close analogy of the modular structure of the agent based model with the schema of the real EA. Thus, the states of the model corresponds to the states of the real process and the model allows to analyze the system in the whole. Consequently, many times new escape trajectories (ways, roads) from EA can be found - especially in more complicated areas. Such an approach can be utilized especially at designing the areas (e.g. like buildings) and to construct them safely in order to avoid crisis situation as soon as possible. On the other hand, at this approach the number of states is usually big, what is a disadvantage of such a procedure. The advantage of the latter approach (i.e. PN-based workflow) is that it yields simpler models with the less number of states. However, on the other hand, the states are not so detailed and in major cases the model is created on the base of existing EA which have to be fully known before. Both approaches can be combined if it is necessary, of course. Thus, their advantages can complement each other.

It is still necessary to emphasize the importance of the supervisors. They bring several improvements, especially: (i) the higher quality of the evacuation process; (ii) more deeply specifying such a process in details at simultaneous repressing lesser-important facts; (iii) simultaneous decreasing the number of states; (iv) they make the exact checking of the expected results possible. Because no graphical PN tool is able to synthesize supervisor(s) the MATLAB tool was used in our research.

References

1. Čapkovič, F.: Modelling, Analysing and Control of Interactions among Agents in MAS. Computing and Informatics 26, 507–541 (2007)
2. Čapkovič, F.: DES Control Synthesis and Cooperation of Agents. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS (LNAI), vol. 5796, pp. 596–607. Springer, Heidelberg (2009)
3. Čapkovič, F.: A Modular System Approach to DES Synthesis and Contro. In: 2009 IEEE Conference on Emerging Technologies & Factory Automation - ETFA 2009. CD ROM, 8 pages. IEEE Press, Piscataway (2009)
4. Čapkovič, F.: Automatic Control Synthesis for Agents and their Cooperation in MAS. Computing and Informatics 29, 1045–1071 (2010)
5. Čapkovič, F.: Cooperation of Agents in Manufacturing Systems. In: Jędrzejowicz, P., Nguyen, N.T., Howlet, R.J., Jain, L.C. (eds.) KES-AMSTA 2010. LNCS, vol. 6070, pp. 193–202. Springer, Heidelberg (2010)
6. Čapkovič, F.: Supervision in Synthesis of Agents Cooperation. International Journal of Information Science 1(1), 24–31 (2011)
7. Iordache, M.V., Antsaklis, P.J.: Supervision Based on Place Invariants: A Survey. Discrete Event Dynamic Systems 16, 451–492 (2006)
8. Iordache, M.V., Antsaklis, P.J.: Supervisory Control of Concurrent Systems: A Petri Net Structural Approach. Birkhauser, Boston (2006)
9. Iordache, M.V.: Methods for the supervisory control of concurrent systems based on Petri nets abstraction. Ph.D. dissertation, University of Notre Dame, USA (2003)
10. Lino, P., Maione, G.: Applying a Discrete Event System Approach to Problems of Collective Motion in Emergency Situations. In: Klingsch, W.W.F., Rogsch, C., Schadschneider, A., Schreckenberg, M. (eds.) Pedestrian and Evacuation Dynamics 2008, pp. 465–477. Springer, Heidelberg (2010)
11. Murata, T.: Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE 77(4), 541–580 (1989)
12. van der Aalst, W.M.P.: Three Good reasons for Using a Petri-net-based Workflow Management System. In: Wakayama, T., Kannapan, S., Khoong, C.M., Navathe, S., Yates, J. (eds.) Information and Process Integration in Enterprises: Rethinking Documents, ch. 10, pp. 161–182. Kluwer Academic Publishers, Boston (1998)
13. van der Aalst, W.M.P., van Hee, K.M., ter Hofstede, A.H.M., Sidorova, N., Verbeek, H.M.W., Voorhoeve, M., Wynn, M.T.: Soundness of Workflow Nets: Classification, Decidability, and Analysis. Formal Aspects of Computing 23(3), 333–363 (2011)
14. Wang, J., Rosca, D., Tepfenhart, W., Milewski, A.: Incident Command System Workflow Modeling and Analysis: A Case Study. In: Van de Walle, B., Turoff, M. (eds.) Proceedings of the 3rd International ISCRAM (Information Systems for Crisis Response and Management) Conference, Newark, NJ (USA), pp. 127–136. ISCRAM Association, Brussels (2006)

Web Service Compositions Which Emerge from Virtual Organizations with Fair Agreements*

Romina Torres, Denise Rivera, and Hernan Astudillo

Universidad Técnica Federico Santa María
Av. España 1680, Valparaíso, Chile
{romina,drivera,hernan}@inf.utfsm.cl
http://www.toeska.cl, http://www.inf.utfsm.cl

Abstract. By wrapping services as active software agents, constantly bidding to consumers' requests, self-organizing web service compositions can become real. But they still lack a negotiation capability. In this paper, we add to our Multiagent component composition system (MACOCO) the negotiation behavior of non-functional requirements, by implementing a modified version of the Zeuthen strategy, which allows both parties to get a fair agreement without knowing the utility function of the counterpart. We study the tradeoff of extra overhead versus a fair agreement, showing the feasibility and performance of this approach. A web prototype tool is available.

Keywords: multiagent systems, service oriented architecture, recommender system, negotiation, Zeuthen strategy, web service composition.

1 Introduction

Service-based systems and the ever increasing proliferation of available services in the Web have drastically changed the way people build Software. Now the problem has changed from building each part from scratch to discover and select a set of already implemented and ready-to-use services. According to Seekda[1] there are more than 28000 available WSDL-based services. Then, for each kind of functionality there is a set of functionally equivalent Web services candidates to be selected as a composition to satisfy the required functionality [WSBCB1].

In [TAC1], we pointed out that, historically, the discovery model of services has been delegated to consumers who have to be capable to decompose their requests in multiple sub-requirements, evaluate different alternatives for each one, and construct their solutions by building compositions that maximize the original request. Unlike web pages, services are dynamic self-contained entities, which can communicate and even collaborate with others in order to help to build new applications; providing functionalities with different quality levels.

* UTFSM PIIC and CCTVal (FB.02PG.11 - Basal FB0821).

[1] http://webservices.seekda.com

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 34–43, 2012.

They should not exhibit the passive nature of web pages. As in economical markets, where providers compete with each other and create new coalitions to gain contracts, services are not so far different from this reality, but on a different scenario. We proposed a different approach, MACOCO, to discover service compositions that naturally arise from the market satisfying complex requirements published by the customers [TRA1]. Because most of the open Web catalogs provide poor information about non-functional aspects (reason why in our previous work we proposed to use an external Quality of service (QoS) Certifier [TAS1]), MACOCO did not consider negotiable aspects. But today, there is an increasing trend of online services that are opening their systems through the implementation of open Web APIs. Large companies like Google, Amazon and eBay have provided interfaces to many of their services at little or no cost, allowing individuals and other businesses to create Web composite applications (mashups). They provide different aspects, potentially negotiable, as *number of concurrent connections or users*, *storage size*, *price*, to name a few.

This article extends the MACOCO's approach allowing customers to negotiate with the n better offers their negotiable aspects using the modified Zeuthen strategy [Wo1] which warranties that both, service providers and customers, to get a fair agreement.

The remaining of this article is organized as follows: Section 2 describes previous related work to address this problem; Section 3 presents an overview of MACOCO framework; Section 4 describes the negotiation behavior; Section 5 introduces MACOCO prototype; Section 6 describes an empirical case study to assess the approach feasibility; and Section 7 summarizes and concludes.

2 Related Work

Zulkernine and Martin [FM1] propose a trusted negotiation broker framework that allows consumers to reach service level agreements with providers using adaptive and intelligent bilateral bargaining process. Agents can change their negotiation strategies, as well their own representation (their price can change as well their QoS). On one hand, they proposed an invasive procedure to gather service providers' QoS measurements that implies to set a module at providers environments; and on the other hand, the adaptability could be dangerous in terms of the convergence of the negotiation, because agents can change strategies during the negotiation, as well as their QoS.

In [BUMST1] and [MS1] they also proposed a framework to discover and select services for non complex requests as well as negotiate agreements between the different parts. Unfortunately, they did not address the service composition problem. In [CWSM1] they also proposed a multi-agent negotiation based method to solve the service composition problem. Unfortunately, they do not provide experiments to prove if their approach is feasible, the utility function does not allow to assign different importance to quality constraints and they do not provide a prototype that people can use.

3 An Overview of MACOCO Framework

MACOCO is a multiagent component composition framework which serves as a recommender system for Software architects. It allows them to obtain service recommendations given a complex request. Architects publishes their requests

Algorithm 1. Utility function representing the request of the software architect

```
Architect provides a description of the request $R_X$. Its utility function is calculated
   as $U_X = \sum_{i=1}^{I} v_i \sum_{j=1}^{J} w_j \delta(c_j)$
for $i = 1$ to $i = I$ do
   Architect specifies the relative importance $v_i$ of each one
   Architect specifies the functional category of the sub-requirement $FR_i$ or provides keywords that allows MACOCO to determine it
   for $j = 1$ to $j = J$ do
      Architect specifies the relative importance of $NFR_j$, $w_j$ .
      Architect specifies the reserved and desired value for $NFR_j$, $R_j$ and $D_j$ respectively ($D_j \neq R_j$)
      if $R_j < D_j$, which means $NFR_j$ is a much is better type of constraint (for instance "reliability" ) then
         $\delta(c_j) = \begin{cases} 0 & \text{if } c_j < R_j \\ \frac{c_j - R_j}{D_j - R_j} & \text{if } R_j \leq c_j \leq D_j \\ 1 & \text{if } c_j > D_j \end{cases}$
         where $c_j$ is the current value provided by the potential candidate
      else
         $\delta(c_j) = \begin{cases} 0 & \text{if } c_j > R_j \\ \frac{R_j - c_j}{R_j - D_j} & \text{if } D_j \leq c_j \leq R_j \\ 1 & \text{if } c_j < D_j \end{cases}$
      end if
   end for
end for
```

which are comprised of functional and non-functional requirements (FR and NFR respectively), their relative importance, and for each NFR its *desired* and *reserved* numerical value. The *reserved* value represents the worst acceptable case, where its utility is almost 0. Each request R_X is represented by an utility function. The composition of services that maximizes this function will be granted with the contract. Algorithm 1 explains how the utility function for the client is built given the request. The utility function for providers is analogously built except for those not negotiable aspects, where they behave as a boolean function (is *greater or equal* or *lesser than* depending of the metric), and for the negotiable aspects they have a δ function exactly opposed to the client. The key element of MACOCO is the *blackboard*, where all the requirements are published. The *blackboard* has a publisher-subscriber architecture with many subscription topics as functionalities types exist. Service providers can voluntarily register their services as well their preferences, marking as negotiable the NFRs that actually are. According to which functionality(ies) the service provides, it is added

Algorithm 2. Web service composition

Require: Given the blackboard B with several sets of service agents SA
Require: Given a request R_X with C FRs, and utility function U_X
The RA Agent Factory wraps the request R_X into the agent RA_X.
RA_X publishes a call for tender at T on the blackboard into a set of categories C during a timeframe t_X. RA_X is suspended until $T + t_X$ has passed.
for $i = 1$ to $i = I$ (where I is the number of SAs subscribed to the C categories where the R_X was published) **do**
  if SA_i partially satisfies the set of FRs **then**
    if SA_i decides to create a VO **then**
      for $k = 1$ to $k = K$; K are of FRs which SA_i not satisfies **do**
        RA Agent Factory creates a minion agent for the subFR_k, RA_k
        RA_k publishes a call for tender expiring at $T + t_Y$ (where $t_Y = t_X * \frac{1}{K}$).
      end for
      if $currentTime \geq (T + t_Y)$ **then**
        for $k = 1$ to $k = K$ **do**
          RA_k recovers the offers and evaluates them with $U_k = \sum_{j=1}^{J} w_j \delta(c_j)$
          RA_k notifies SA_i the offer's selection
        end for
      end if
      if SA_i decides to make a bid as a VO) **then**
        The Agent Factory creates the VO_i agent instance
        The VO_i sends the offer O_i into the queue of the RA_X
      else
        SA_i sends the offer O_i into the queue of the RA_X
      end if
    end if
  end if
end for
if $currentTime \geq (T + t_X)$ **then**
  RA_X recovers the offers and evaluates them using U_X.
  Creates a contract C_X between the RA_X and the highest ranked offer
end if

into the subscription topic(s) of the blackboard. Then, next time that a request is published in the topic of the blackboard in which this service is subscribed, it will be notified.

When a service provider is registered, MACOCO creates an instance of a service agent SA_i which is subscribed to the topic j of the blackboard. For each request, MACOCO creates an instance of a requester agent (RA_X). It determines which are the functionalities requested in order the agent RA_X publishes a request for tender on each corresponding subscription topic of the blackboard (with a deadline to receive offers). When RA_X publishes its request the service composition process starts, as summarized in Algorithm 2. The request is published in different functional categories of the blackboard. Each agent subscribed to these categories makes the decision to bid or not. If the service agent satisfies only X% of the required functionality, it can bid directly, else it can create a virtual organization to find partners which complement it. Successful VOs (which have been created to bid specific requests) are registered as another provider agent in order to serve new similar requests in the future. If a service agent (or VO) is granted with a contract, then both parts are properly notified. Unfortunately, until now MACOCO did not allow consumers and providers to make deals on those negotiable aspects, meaning that typically all the offers and generated contracts are the best deal for providers using the *desired* value for each non-functional aspect, such as price. This could be inconvenient even for providers, because if the provider was willing to relax not critical aspects, it could obtain the contract.

Algorithm 3. RA_X setting up the negotiation with the n best offers

Require: The n higher ranked offers before negotiation
1: **for** $i = 1$ to $i = n$ **do**
2: RA_X calculates its preference vector P_X such that $P_X \leftarrow argmax_P U_X(P)$. At the beginning the vector P_X has for all the negotiable aspects the desired values.
3: RA_X proposes to offerer i P_X
4: **end for**

4 Adding Negotiation Capabilities to the MACOCO Framework

In this Section we extends MACOCO to allow service and request agents negotiate non-functional aspects obtaining better deals for both parts. We have chosen a slightly modified version of the Zeuthen negotiation strategy because of its properties (it is Pareto optimal and it maximizes the Nash product).

In the original MACOCO, the RA_X granted the contract to the provider with the highest ranked offer. Now, instead of just accepting the best offer, it negotiates with the top n offers. Then, the RA_X sends to the responsible of these offers (service or VO agents) a counteroffer modifying the values (from the *desired* value to the *reserved* value) of those aspects which are negotiable for both parts. To force first negotiations, the requester agent sends its preferred deal with $risk = 1$, so the provider agent is force to concede (if it does not accept the deal in the first place). Algorithm 3 shows the negotiation setup.

At each round of the negotiation process, the agent receiving the counteroffer (service or request agent) evaluates the utility that it would obtain by accepting the proposal of the counterpart. As we mentioned before, the utility function of requesters is exactly inverse to services. For instance, for the *price* aspect, requesters prefer a cheaper price, contrarily to providers which prefer a expensive one. Then, if requesters specify an aspect with a *desired* and *reserved* value in such way that $R_j < D_j$, then for providers this aspect should be $D_j < R_j$.

Algorithm 4. Slightly modified version of the Zeuthen-monotonic-concession protocol

Agent j proposes P_j
if Agent j accepted or refused offer P_i **then**
    Close negotiations and calculate $U_i(P_i)$ if ACCEPT_PROPOSAL; otherwise stay with conflict deal.
end if
if $U_i(P_j) \geq U_i(P_i)$ **then**
    Accept P_j, send ACCEPT_PROPOSAL. Close negotiations.
end if
$risk_i \leftarrow \frac{U_i(P_i)-U_i(P_j)}{U_i(P_i)}$
if $risk_i \leq risk_j$ **then**
    if all $c_i = R_i$ **then**
        no new offer can be made, send REJECT_PROPOSAL. Close negotiations and stay with conflict deal.
    else
        calculates $P_i \leftarrow P_i'$ which concedes a little for each negotiable aspect moving a shorter step from *desired* to *reserved* values
        while $c_i < R_i$ if $R_i > D_i$; or $c_i > R_i$ if $R_i < D_i$ **do**
            add (or substract, depending the metric) 10% of the difference between R_i and D_i to new offer c_i'
        end while
        calculate new $risk_i(P_i')$
        send new P_i' with $risk_i(P_i')$
    end if
else
    no need to concede, send same offer P_i
end if
wait for counteroffer or an ACCEPT_PROPOSAL or REJECT_PROPOSAL
goto Step 1

It is important to notice the offers are not simultaneously sent as the Zeuthen strategy dictates. Since we are removing the need for the other agent to know the utility function of its counterpart, we are also simplifying the need for synchronization. At each negotiation round, each agent calculates its own risk, and informs it to the counterpart; that way the other can compare and decide if its turn to concede or not. For this to work, it is important that the negotiations to be an offer-and-counteroffer type; otherwise the agents will fail to inform its own updated risk and no comparison can be made.

The agent calculates its own utility with the offer of the counterpart, and also measures its risk. It compares the known risk of the counterpart and its own. The one with less risk should concede just enough so that it does not have to concede again in the next round. Then, the agent which concedes must determine how much it should concede. Instead of use the original idea of Zeuthen which demands that agents knows the utility function of each other, we use a simple heuristic that basically adds 10% of the difference between *reserved* and *desired* values to all negotiable aspects (of course this parameters should be empirically determined). In this way the concession is almost always sufficient to invert the risks and force the other agent to concede on the next round. Then, the offers are sent and a new negotiation round starts again. If the concession is not enough to invert the risks, the other agent simply sends the same offer back and forces the other one to re-evaluate and to think of a new offer. If no new offer can be made (meaning the agent reach all its *reserved* values on the offers), it ends the negotiations, staying with the conflict offer. Algorithm 4 summarizes the negotiation procedure.

The Zeuthen strategy is guaranteed to terminate and the agreement it reaches upon termination is guaranteed to be individually rational and Pareto optimal.

5 Implementation

MACOCO is implemented in Java over JADE [2] (Further details of the implementation can be found in [TRA1]). In this paper we added to MACOCO, and specifically to agents the negotiation capability as a new behavior. It was implemented in a flexible way in order to allow the strategy could be interchangeable with other by implementing a Java interface. All the functionality of the MACOCO are exposed as REST services. A Web prototypal tool [3] is available.

6 Validation

MACOCO is set up with a service market of 1500 operative Web services which is a subset of the original QWS Dataset [AM1] [4] which originally included 2507 actual Web service descriptors with nine QoS measurements. The non-functional aspects are *response time*, *availability*, *throughput*, *successability*, *reliability*, *compliance*, *best practices*, *latency* and *documentation*. All the experiments were executed over a personal notebook, model Dell vostro 1400, Core 2 Duo T5470 (1.6Ghz, 2MB L2 Cache, 800MHz FSB), 2GB DDR2 667Mhz of RAM, and 60GB 5400RPM SATA of hard drive.

We have synthetically added the price because it is nor part of the original dataset. We calculate a price which is directly proportional to the quality of the service. For each functional category and for each quality aspect we classified the services in five quality levels. Depending of which level the service belongs, it gains a number of points (between 1 and 5, where 1 is the worst level and five the best). Potentially, for each category, the best service could gain 45 points and the worst service could gain 9 points. Then, we calculate the corresponding price in order to get the relation between the quality and price: "as the quality increases the prices also increases". Only the price aspect has been marked as a negotiable aspect.

[2] http://jade.tilab.com/

[3] http://cc.toeska.cl/broker

[4] http://www.uoguelph.ca/~qmahmoud/qws

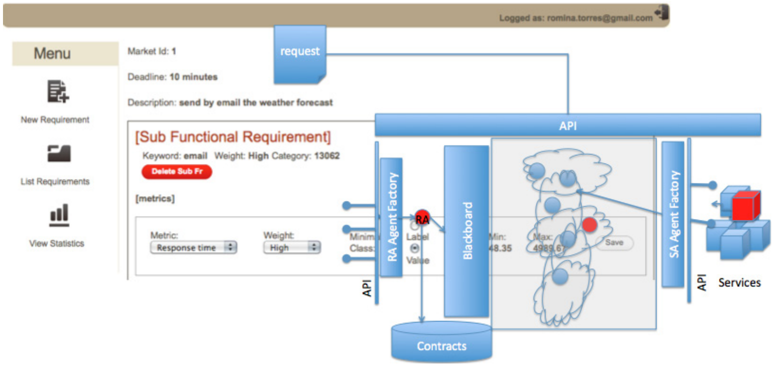


Fig. 1. MACOCO prototype

Further details of the scalability of MACOCO with concurrent requests can fe found in [TRA1]. In this work we study the relation between the overhead produced by the negotiation and the extra utility gained by the customers and how this overhead increases according to the number of offerers with which the client negotiates.

6.1 First Experiment

The aim of this experiment is to measure (for a set of ten requests with one sub-FR and with up to five QC) how the negotiation process ensures, to client and providers, to obtain fairer agreements to both parts. We set n to 1. Given the initial agreement, customers and providers discuss how this agreement could be fairer for both, using the modified Zeuthen strategy. Table 1 shows the results. The first and second column indicates the id of the test case and the R and D price of the RA agent. The third column shows the initial and final utility of the RA agent and the fourth shows the same for the SA agent. The last three columns show the overhead in seconds produced by the negotiation process, the number of rounds in which the negotiation converges to fair agreements and the variation of price from initial to final one. In seven of the ten cases the negotiation was successful. For instance in the test #1 the RA agent chose the offer of the SA agent offered to it to 42. Then, we forced the negotiation making the RA agent to send a counteroffer to the PA agent and after of 7 rounds the utility of the RA agent increases a 14% with an overhead of almost 1 second. Of course, the utility of the provider SA drops from 1 to 0.85. The price which starts from the 42 drops to 39.3 which is not exactly the ideal price of the RA agent but was a fair agreement for both. Now, three of the ten cases the utility remained the same, then there was not negotiation, because the RA agent does not need to negotiate because it has what it ask.

Table 1. Overhead versus fair agreement

test	$P_{RA_{[R,D]}}$	$U_{RA_{I-F}}$	$U_{SA_{I-F}}$	O[s]	#rounds	P_{I-F}
1	[46.2–38.4]	[0.8–0.94]	[1 - 0.85]	0.7	7	[42–39.3]
2	[50–42]	[0.91 - 0.91]	[1 0.98]	2.1	14	[40–39.2]
3	[83–69]	[0.82 - 0.82]	[1 - 0.975]	0.7	3	[60–58]
6	[78–65]	[0.97 - 0.98]	[1 - 0.98]	0.5	5	[66–64.6]
7	[71–59]	[0.98 - 0.98]	[1 - 0.967]	1.2	5	[56–54.4]
10	[47.5–39.6]	[0.81–0.9]	[1–0.9]	1.4	8	[44–42]

6.2 Second Experiment

The aim of this experiment is to show how the overhead increases as n (the number of offerers with which the client negotiates) increases (See Table 2). We perform this experiment with one request at a time.

Table 2. Overhead produced by negotiate with several providers at the same time

#Counterparts	Overhead[s]	#rounds with each offerer	Total of rounds
1	0.78	{5}	5
3	1.47	{3,5,9}	17
5	3.4	{3,4,5,5,5}	22
10	4.2	{3,3,3,4,4,4,4,5,7,8}	45

The application scales. It is safe to encourage clients to allow their representing agents to negotiate with several offerers at the same time.

7 Conclusions

This article is the continuation of our initial work, MACOCO, a framework capable to obtain web service compositions from the market which naturally arise from it and addresses the request. In this article we have show how clients can negotiate with service providers those aspects that are negotiable. We have showed that even when there is an overhead proportional to the number of offerers with the client negotiates, the framework scales when the number of concurrent requests increases and that the extra utility that the client obtains by negotiation is enough to accept the extra overhead. By using a slightly modified version of the Zeuthen strategy, avoiding the need for agents to know the utility function of each other. This way agents obtain fair agreements without risk that one of them could take advantage of the privileged information, such as the minimal values of the counterpart in order to maximize only the utility of one counterpart of the contract.

References

[AM1] Al-Masri, E., Mahmoud, Q.: WSB: a broker-centric framework for quality-driven web service discovery. Software: Practice and Experience 40, 917–941 (2010)

[BUMST1] 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 (AAMAS 2009), vol. 2, pp. 1395–1396 (2009)

[CWSM1] Cao, J., Wang, J., Zhang, S., Li, M.: A multi-agent negotiation based service composition method for on-demand service. In: 2005 IEEE International Conference on Services Computing, July 11-15, vol. 1, pp. 329–332 (2005)

[FM1] Zulkernine, F., Martin, P.: An Adaptive and Intelligent SLA Negotiation System for Web Services. IEEE Transactions on Services Computing 4(1), 31–43 (2010)

[MS1] Mahbub, K., Spanoudakis, G.: Proactive SLA Negotiation for Service Based Systems: Initial Implementation and Evaluation Experience. In: 2011 IEEE International Conference on Services Computing (SCC), pp. 16–23 (2011)

[TAC1] Torres, R., Astudillo, H., Canessa, E.: MACOCO: A Discoverable Component Composition Framework Using a Multiagent System. In: IEEE Proceedings of the XXIX International Conference Chilean Computer Science Society, pp. 152–160 (2010)

[TAS1] Torres, R., Astudillo, H., Salas, R.: Self-Adaptive Fuzzy QoS-Driven Web Service Discovery. In: Proceedings IEEE International Conference on Services Computing (SCC 2011), pp. 64–71 (2011)

[TRA1] Torres, R., Rivera, D., Astudillo, H.: From virtual organizations to Self-organizing Web service compositions. To appear on the IEEE Proceedings of the XXIX International Conference Chilean Computer Science Society (2011)

[Wo1] Wooldridge, M.: An Introduction to Multiagent Systems. John Wiley and Sons (2002)

[WSBCB1] Whittle, J., Sawyer, P., Bencomo, N., Cheng, B., Bruel, J.: RELAX: a language to address uncertainty in self-adaptive systems requirement. In: Requirements Engineering. Springer, London (2010) ISSN 0947-3602

Evaluation of an Intelligent Tutoring System Incorporating Learning Profile to Determine Learning Scenario*

Adrianna Kozierkiewicz-Hetmanska

Institute of Informatics, Wroclaw University of Technology, Poland
adrianna.kozierkiewicz@pwr.wroc.pl

Abstract. The modern teaching methods should allow effective learning in a short time. Development and detailed analysis of an e-learning system as well as a research on influence of the applied didactic methods on effectiveness of the learning process is required. Therefore, this paper is devoted to presentation of the results of an experiment conducted using prepared prototype of an e-learning system with a personalized courseware. Authors run researches on an influence of a personalization on the learning process effectiveness. Significantly higher results were obtained by learners whose user profiles were taken into consideration during the determination of the learning scenario. This conclusion is true when the learning material is presented for the first time, when users are underaged, male or have had elementary education.

Keywords: intelligent tutoring system, statistical analysis, learning scenario.

1 Introduction

The rapid change on technological fields in recent years bring about that the creation, distribution, diffusion, use, integration and manipulation of information is a significant economic, political, and cultural activity. Characteristics of information society is the large and broadly available amount of information, the mobility of people, the attention to optimization, the aspiration to increase the number of workplaces, the change of work into telework according to lean manufacturing model and especially the change in the education sector.

The modern teaching methods should differ from methods which have been used several years ago. Students need to learn faster, more efficiently and consolidate knowledge so that it is not forgotten despite passage of time. The key to educational success seems to be the e-learning and a possibility to individualize the learning process. The intelligent tutoring system should provide a personalized learning material suitable for students' current knowledge level, needs, preferences, learning styles, abilities and interests.

* This research was financially supported by Polish Ministry of Science and Higher Education under the grant N N519 407437.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 44–53, 2012.

In the previous work author proposed the model of the intelligent tutoring system which offers an individual learning process on each step. Firstly, the system collects information about a student during the registration process [8]. Next, the user is classified to a set of similar learners. The criterion of classification is a set of attributes selected by an expert. The opening learning scenario for a new learner is determined based on finished scenarios of students who belong to the same class as the new one. The algorithms for choosing the opening learning scenario use consensus theory. The methods for determination of an opening learning scenario with a proposed knowledge structure and a learning scenario are described in [9], [13] and [14]. After generating the opening learning scenario the student is presented with the first lesson. When he finishes, he has to pass a computer adaptive test where each question is selected based on the current student's knowledge level [11]. The result of test implicates the next step. If a student achieves a sufficient test score (more than 50%) he continues learning according to the previously selected learning scenario. Otherwise, system suggests relearning. Author proposed two methods one of which has been applied with Bayesian network. The methods of modification of a learning scenario are described in [13], [14], [12] and [10]. The learning process is finished if all lessons from the learning scenario are taught.

The simplified version of the intelligent tutoring system described above was implemented and used to conduct an experiment. In this paper we want to introduce the results of an experiment which are expansions of research presented in [15]. The aim of our work is to prove the hypothesis that it is possible to increase the effectiveness of the intelligent tutoring systems by taking into consideration the user profile in determination of a learning scenario. We investigate the influence of personalization of the learning process according to sex and educational level. Additionally, we study the learning scenario which was the most popular among others and which was giving the best results.

In the next chapter an overview of intelligent tutoring systems and assessment of their efficiency are presented. Section 3 describes the concept of the experiment and presents the results of the experiment with appropriate conclusions. Finally, general conclusions and future work are described.

2 Related Works

In 1982, Sleeman and Brown first coined the term intelligent tutoring systems to describe these evolving systems and distinguish them from the previous CAI systems [17]. From this time many different intelligent tutoring systems were modeled where various techniques such as: Bayesian network, neural network, fuzzy logic, ontologies etc. were applied. Furthermore, proposed systems offer different degree of the adaptation.

System ELM-ART [18] is one of the first and adaptive web-based system. System offers an individual curriculum sequencing and an adaptive annotation of links. Additionally, the system supports an adaptive testing (based on the simple algorithm which selects next question) and stores student's preferences concerning the form of educational pages and uses this information for creating a suitable learning environment. The researches pointed out that system ELM-ART is more effective than his previous version: ELM-PE.

System INSPIRE [5] throughout interaction with the learner dynamically generates lessons. Each lesson is adopted to the learner's knowledge level, learning style (according to Honey & Mumford theory) and follows his progress. The system supports learner's navigation and orientation in the lesson contents by annotating the links. INSPIRE offers computer adaptive testing based on Item Response Theory.

Shikshak [3] plans the individual teaching method based on user's performance level and selects the type of material (such as explanation based, visual based, instructional) as suited to the learning style of a particular student using fuzzy rules. The system showed some improvement (overall 4%) in the performance compared to the performance of students learning in standard classrooms.

System SEDHI [16] supports adaptive navigation by hiding and annotating links. System classifies the students according to profiles that were defined based on statistical study and offers the suited navigation. Preliminary research of this system demonstrated the appropriateness of the applied methods.

The adaptive presentation and navigation is also supported by system ILASH [1] and DEPTHS [7]. Authors of system DEPTHS showed that students who learned with DEPTHS performed better than students who learned in the traditional way.

An interesting solution was proposed in [2], where authors describe a multi-agent e-learning system. The system, which is based on Item Response Theory, offers adaptive tests. One of the agents of the system tries to identify student's learning problems and recommends appropriate learning material to the user. This functionality of the system is obtained by applying an artificial neural network. The experimental research demonstrated that the proposed system can provide personalized and appropriate material course recommendations with the precision of 83.3%.

The model of intelligent tutoring system based on Bayesian network is proposed in [4]. Using the Bayesian network system assesses user's state of knowledge and preferences and determines what the student should do: learn a new topic, learn the current topic thoroughly, repeat the previous topic or shallow scan a new topic.

In our previuous work we proposed procedures allow offering an optimal learning path in each step of an educational process. Some of mentioned systems are not able to offer all functionalities i.e. system SEDHI, ILASH, DEPTHS do not offer adaptive tests. Our method of modification of the learning scenario is innovative because there has been no solution of the problem of modification of a learning scenario during a learning process and based on a current students characteristic. Additionally, the evaluation of intelligent tutoring systems is an important but often neglected stage of an e-learning system development. Very often the resarches are limited to small sample size (i.e. DEPTHS: sample size equals 42, ELM-ART: less than 30) therefore are not statistically significant.

In this work we present the results of experiment using the prototype of the designed system. Our system was examined by a quite big population and the drawn conclusions may be generalized.

3 The Results of Experiment

In our research we want to investigate the influence of learning materials personalization. The research was conducted in the prototype of a e-learning system. In our experiment there were 297 participants. In this group 123 people did not have driving licenses which means that they have learned the presented learning material for the very first time. The learning material concerned the intersections, roadway signs related to intersections and right-of-way laws. Authors prepared 11 learning scenarios which differ from each other in the lessons order and the presentation methods.

Our experiment started with collecting information about a user. Next, the student could be assigned to two groups: a control group or an experimental group based on sex, educational level, age and possession of driving license. Users from the control group were offered a universal learning scenario which is a combination of dedicated learning scenarios. All students from the control group were taught using the same learning scenario. If the student was assigned to the experimental group the system chose a personalized learning scenario suitable for his learning style. After the learning the system evaluated student's learning results. The test consisted of 10 questions chosen randomly from a question bank. The test was supposed to be completed within 10 minutes.

The previous research [15] demonstrated that taking into account student's preferences and learning style for planning the learning scenario increases the effectiveness of the learning process. Students who were offered the personalized learning scenario achieved better learning results by more than 7.182% and less than 7.976% than students who were proposed the universal learning scenario but only in case of the learning material was presented for the first time. Furthermore, the personalization had the statistically significant influence on improvement of the learning result in case of underaged users.

In this paper we want to show the influence of the personalization of the learning process according to sex and education level. The results of the first experiment is presented in Figure 1.

Hypothesis 1. The mean test scores of the experimental and control groups for women satisfies the following assumption: $H_0 : \mu_{exp} = \mu_{cont}$ or the alternative assumption: $H_1 : \mu_{exp} > \mu_{cont}$.

Proof. Firstly, we check the normality of the distribution of analyzed features using the Shapiro-Wilk test. For both groups (experimental and control) we can conclude the normality of distribution of analyzed features ($W_{exp} = 0.9001 > W_{(0.05,17)} = 0.892$ and $W_{cont} = 0.922 > W_{(0.05,23)} = 0.914$) and next, we use the parametric test. There is no information considering the standard deviation and sizes of groups are also different. We need to check the equality of the

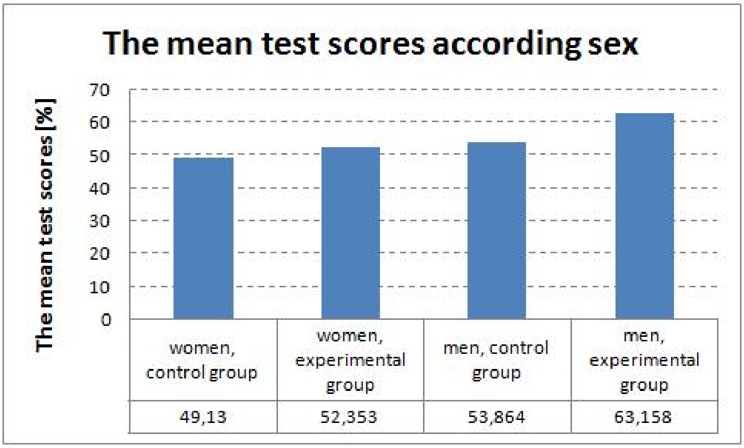


Fig. 1. The mean test scores according sex

standard deviation using F-Snedecor test. We obtain $F = 2.4146$. For $\alpha = 0.05$ the tested statistical value does not belong to the confidence interval $[2,495,\infty)$ (two-tailed test), so we cannot reject the hypothesis that the standard deviation of two samples are equal. For testing the null hypothesis the t-Student test is assumed:

$$t = \frac{\mu_{exp} - \mu_{cont}}{\sqrt{\frac{S^2_{exp}*(n_{exp}-1)+S^2_{cont}*(n_{cont}-1)}{n_{exp}+n_{cont}-2} * (\frac{1}{n_{exp}} + \frac{1}{n_{cont}})}} \tag{1}$$

where: μ_{exp}- average score value for the experimental group, μ_{cont}- average value for the control group, S_{exp}- standard deviation for the experimental group, S_{cont}- standard deviation for the control group, n_{exp}-size of the experimental group, n_{cont}-size of the control group.

The tested statistical value is equal to:

$$t = \frac{52.353 - 49.13}{\sqrt{\frac{24.86^2*(17-1)+16.28^2*(23-1)}{17+23-2} * (\frac{1}{23} + \frac{1}{17})}} = 0.486$$

The significance level of 0.05 is assumed. The confidence interval is equal to $[1.684, \infty)$. The tested statistical value does not belong to confidence interval, that is why the null hypothesis cannot be rejected.

Conclusion 1. *The mean test score of the experimental and control groups are not significantly different in case of female students.*

Hypothesis 2. The mean test scores of the experimental and control groups for men satisfies the following assumption: $H_0 : \mu_{exp} = \mu_{cont}$ or the alternative assumption: $H_1 : \mu_{exp} > \mu_{cont}$.

Proof. In both groups (experimental and control) we cannot reject the hypothesis about the normality of the distribution of analyzed features (the Shapiro-Wilk test: $W_{exp} = 0.9764 > W_{(0.05,38)} = 0.938$ and $W_{cont} = 0.9508 > W_{(0.05,44)} = 0.944$). As it was done before we check the equality of the standard deviation using the F-Snedecor test. We obtain $F = 0.7449$. For $\alpha = 0.05$ the tested statistical value does not belong to the confidence interval $[1.88, \infty)$ (two-tailed test), so we cannot reject the hypothesis that the standard deviation of two samples are equal. Next, we calculate the statistic (1):

$$t = \frac{63.158 - 53.864}{\sqrt{\frac{21.411^2 * (38-1) + 24{,}699^2 * (44-1)}{38+44-2} * (\frac{1}{38} + \frac{1}{44})}} = 1.81$$

The significance level of 0.05 is assumed. We get $t \in [1.68, \infty)$, that is why the null hypothesis is rejected and the alternative hypothesis is assumed.

Conclusion 2. *The mean test score of the experimental group is greater than the mean score of the control group in case of male students.*

In the second stage we verify the influence of the personalization of the learning process according to education level. The results are presented in Figure 2.

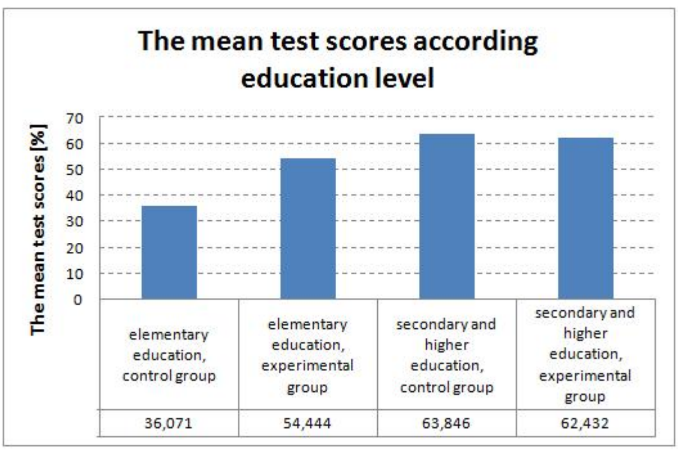


Fig. 2. The mean test scores according education level

Hypothesis 3. The mean test scores of the experimental and control groups for people with the elementary education satisfies the following assumption: $H_0 : \mu_{exp} = \mu_{cont}$ or the alternative assumption: $H_1 : \mu_{exp} > \mu_{cont}$.

Proof. Firstly, we check the normality of the distribution of analyzed features using the Shapiro-Wilk test. For both groups (experimental and control) we can conclude the normality of distribution of analyzed features ($W_{exp} = 0.9628 > W_{(0.05,18)} = 0.897$ and $W_{cont} = 0.9359 > W_{(0.05,28)} = 0.924$). Using F-Snedecor test we obtain F=1.7905. For $\alpha = 0.05$ the tested statistical value does not belong to the confidence interval $[2.33, \infty)$ (two-tailed test), so we cannot reject the hypothesis that the standard deviation of two samples are equal. For testing the null hypothesis the t-Student test is assumed (1):

$$t = \frac{54.44 - 36.071}{\sqrt{\frac{21.915^2 * (18-1) + 16.548^2 * (28-1)}{18+28-2} * (\frac{1}{18} + \frac{1}{28})}} = 3.23$$

The significance level of 0.05 is assumed. We get $t \in [1.68, \infty)$. The null hypothesis is rejected and alternative hypothesis is assumed.

Conclusion 3. *The mean test score of the experimental group is greater than the mean score of the control group in case of elementary school graduates only.*

Hypothesis 4. The mean test scores of the experimental and control groups for people with the secondary and higher education satisfies the following assumption: $H_0 : \mu_{exp} = \mu_{cont}$ or the alternative assumption: $H_1 : \mu_{exp} > \mu_{cont}$.

Proof. In both groups (experimental and control) we cannot reject the hypothesis about the normality of the distribution of analyzed features (the Shapiro-Wilk test: $W_{exp} = 0.9447 > W_{(0.05,37)} = 0.936$ and $W_{cont} = 0.9399 > W_{(0.05,39)} = 0.939$). The value of F-Snedecor test equals F=1.6239 and does not belong to the confidence interval $[1.88, \infty)$ (two-tailed test), so we cannot reject the hypothesis that the standard deviation of two samples are equal. As previously we calculate the following statistic:

$$t = \frac{62.432 - 63.846}{\sqrt{\frac{23.18^2 * (37-1) + 18.20^2 * (39-1)}{37+39-2} * (\frac{1}{37} + \frac{1}{39})}} = -0.297$$

For the significance level equal to 0.05 we get $t \notin [1.665, \infty)$. The null hypothesis could not be rejected.

Conclusion 4. *The mean test score of the experimental and control groups are not significantly different in case of secondary or higher education graduates.*

Both groups of students with and without driving licenses were proposed an interactive-graphical-sequential version of the learning scenario most often. It means that users who took part in the experiment were mostly visual, active and sequential regarding the learning style. Rarely, students had the strong preference in any dimension of the receiving information. The Figure 3 presents the number of offered learning scenario.

It turned out that students without the driving license achieved the best learning results if they were presented an interactive-graphical-sequential version of the learning scenario. Users possessing the driving license preferred an

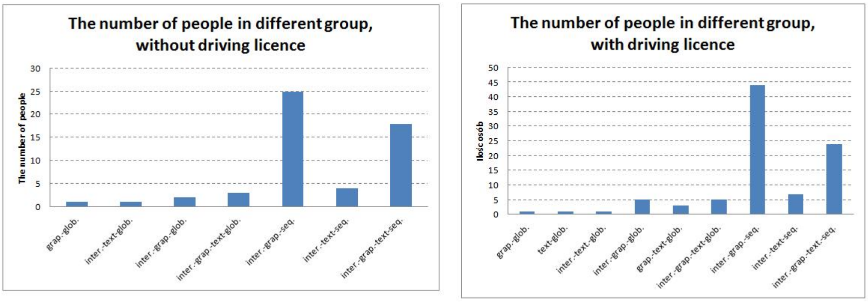


Fig. 3. The number of people in different groups

interactive-graphical-global version of the learning scenario. In both groups (with and without the driving license) students got a bit poorer test scores if they were offered an interactive-graphical-text-sequential version of the learning scenario. Those conclusions were drawn only for versions of the learning scenario which were offered to at least 5 users.

4 Conclusion

The personalization of the learning scenario increases the effectiveness of the learning process. The previous research demonstrated that students achieved better learning results if they were offered the learning scenario suitable for their learning style but only in case of students who have not learned the presented learning material before. Furthermore, the personalization have the statistically significant influence on the improvement of the learning result in case of underaged users. In this paper we extend our research and test the influence of personalization according to sex and educational level. Taking into account the learning profile to determine the learning scenario have statistically significant influence on the improvement of the learning results if students are male or have the elementary education.

In the further work we are planning to design the new methods which could increase the effectiveness of the e-learning systems. First of them is an algorithm of an intelligent class monitoring. In the e-learning systems it is needed to distinguish better from worse students. Better students need more complicated tasks, more details and curiosities. Worse users demand more time for solving tasks, more repetitions and hints. The results of described methods should be considered in the determination of the learning scenario. The second method refers to offering a collaborative learning. Algorithms should create a group of students and offer them suitable tasks. It is also planned to conduct the experimental tests using the prototype of the e-learning system with both algorithms implemented.

References

1. Bajraktarevic, N., Hall, W., Fullick, P.: ILASH: Incorporating learning strategies in hypermedia. In: Proc. 14th Conf. Hypertext and Hypermedia, pp. 145–154 (2003)
2. Baylari, A., Montazer, G.A.: Design a personalized e-learning system based on item response theory and artificial neural network approach. Expert Systems with Applications 36, 8013–8021 (2009)
3. Chakraborty, S., Roy, D., Basu, A.: Shikshak: An Architecture for an Intelligent Tutoring System. In: Proc. 16th International Conference on Computers in Education, Taipei, Taiwan, pp. 24–31 (2008)
4. Gamboa, H., Fred, A.: Designing intelligent tutoring systems: a bayesian approach. In: ICEIS 2001, Artificial Intelligence and Decision Support Systems, pp. 452–458 (2001)
5. Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H., Magoulas, G.D.: INSPIRE: An INtelligent System for Personalized Instruction in a Remote Environment. In: Reich, S., Tzagarakis, M.M., De Bra, P.M.E. (eds.) OHS/SC/AH 2001. LNCS, vol. 2266, pp. 215–225. Springer, Heidelberg (2002)
6. Felder, R.M., Silverman, L.K.: Learning and Teaching Styles in Engineering Education. Engineering Education 78(7), 674–681 (1988)
7. Jeremic, Z., Jovanovic, J., Gašević, D.: Evaluating an Intelligent Tutoring System for Design Patterns: the DEPTHS Experience. Journal of Educational Technology & Society 12(2), 111–130 (2009)
8. Kozierkiewicz, A.: Content and structure of learner profile in an intelligent E-learning system. In: Nguyen, N.T., Kolaczek, G., Gabrys, B. (eds.) Knowledge Processing and Reasoning for Information Society, pp. 101–116. EXIT Warsaw (2008)
9. Kozierkiewicz, A.: Determination of Opening Learning Scenarios in Intelligent Tutoring Systems. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) New Trend in Multi-media and Network Information Systems, pp. 204–213. IOS Press (2008)
10. Kozierkiewicz-Hetmańska, A.: A Conception for Modification of Learning Scenario in an Intelligent E-learning System. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS (LNAI), vol. 5796, pp. 87–96. Springer, Heidelberg (2009)
11. Kozierkiewicz-Hetmańska, A., Nguyen, N.T.: A Computer Adaptive Testing Method for Intelligent Tutoring Systems. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010. LNCS, vol. 6276, pp. 281–289. Springer, Heidelberg (2010)
12. Kozierkiewicz-Hetmańska, A., Nguyen, N.T.: A Method for Scenario Modification in Intelligent E-Learning Systems Using Graph-Based Structure of Knowledge. In: Nguyen, N.T., Katarzyniak, R., Chen, S.-M. (eds.) Advances in Intelligent Information and Database Systems. SCI, vol. 283, pp. 169–179. Springer, Heidelberg (2010)
13. Kozierkiewicz-Hetmańska, A., Nguyen, N.T.: A method for learning scenario determination and modification in intelligent tutoring systems. Applied Mathematics and Computer Science 21(1), 69–82 (2011)
14. Kozierkiewicz-Hetmańska, A.: A Method for Scenario Recommendation in Intelligent E-Learning Systems. Cybernetics and Systems 42(2), 82–99 (2011)

15. Kozierkiewicz-Hetmańska, A.: Evaluating the Effectiveness of Intelligent Tutoring System Offering Personalized Learning Scenario. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ACIIDS 2012, Part I. LNCS, vol. 7196, pp. 310–319. Springer, Heidelberg (2012)
16. da Silva, G.T., Rosatelli, M.C.: Adaptation in Educational Hypermedia Based on the Classification of the User Profile. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 268–277. Springer, Heidelberg (2006)
17. Sleeman, D., Brown, J.S. (eds.): Intelligent Tutoring Systems, pp. 1–11. Academic Press, New York (1982)
18. Weber, G., Peter Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-based Instruction. International Journal of Artificial Intelligence in Education 12, 351–384 (2001)

Strategy-Based Learning through Communication with Humans

Nguyen-Thinh Le and Niels Pinkwart

Clausthal University of Technology,
Germany
{nguyen-thinh.le,niels.pinkwart}@tu-clausthal.de

Abstract. In complex application systems, there are typically not only autonomous components which can be represented by agents, but humans may also play a role. The interaction between agents and humans can be learned to enhance the stability of a system. How can agents adopt strategies of humans to solve conflict situations? In this paper, we present a learning algorithm for agents based on interactions with humans in conflict situations. The learning algorithm consists of four phases: 1) agents detect a conflict situation, 2) a conversation takes place between a human and agents, 3) agents involved in a conflict situation evaluate the strategy applied by the human, and 4) agents which have interacted with humans apply the best rated strategy in a similar conflict situation. We have evaluated this learning algorithm using a Jade/Repast simulation framework. An evaluation study shows two benefits of the learning algorithm. First, through interaction with humans, agents can handle conflict situations, and thus, the system becomes more stable. Second, agents adopt the problem solving strategy which has been applied most frequently by humans.

Keywords: Agent-Human learning, multi-agent systems, machine learning, evaluation.

1 Introduction

In complex application systems, there exist not only autonomous components which can be represented by agents, but humans may also play an important role. Usually, in a multi-agent system, agents have specific pre-defined abilities to perform a certain task. One of the challenges of a multi-agent system is to develop agents with the ability to learn from human behavior. Current research on multi-agent learning exploits machine learning techniques to adapt to preferences or behaviors of human users. In this paper, we present an algorithm that allows autonomous agents to learn problem solving strategies in conflict situations through communication with humans.

Our learning algorithm assumes that humans may have several strategies when encountering a conflict situation. Researchers suggested that experts have some kind of knowledge about problem categories and associated solution strategies

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 54–64, 2012.

which lead to correct solutions [7]. When solving a problem, an expert solver will identify the problem characteristics by associating them with previously solved problems. The problem will be assigned to a solution strategy which can then be applied to solve similar problems. In this paper, the term *strategy* is noted as a particular way of solving a problem as done by human experts. Under this assumption, we propose a strategy-based learning algorithm for autonomous agents. The algorithm consists of four phases. First, the human meets agents in a conflict situation. Then, in the second phase, the human initiates a conversation with involved agents and chooses a strategy to solve the conflict. Third, the agents which are involved in the conflict situation rate the effectiveness of the proposed strategy. In the last phase, based on an aggregated rating score, agents apply the most effective strategy they have learned in similar future conflict situations. To show the benefits of this learning algorithm, we will test the following two hypotheses:

1. Agents applying the strategy-based learning algorithm will adopt the strategy applied most frequently by humans in a similar conflict situation.
2. The more agents applying this algorithm interact with humans in conflict situations, the more stable the system is.

2 State of The Art

There have been numerous attempts on developing algorithms for multi-agent learning. They can be classified into three main approaches which differ from each other in terms of the type of feedback provided to the learner [11]: supervised, unsupervised, and reward-based learning. In supervised learning, the feedback provides the correct output and the objective of learning is to match this desired action as closely as possible. In unsupervised learning, no feedback is provided and the objective is to seek useful activities on a trial-and-error basis. In reward-based learning, the feedback provides a quality assessment (the *reward*) to the learner's action and the objective is to maximize the reward. The class of reward-based learning approach is divided into reinforcement learning methods (which estimate value functions) and stochastic search methods (which directly learn behaviors without explicit value functions). Among the three learning approaches, the reward-based one is most frequently used to enable agents to learn from experience. The supervised learning and unsupervised learning approaches may be difficult to develop. Reinforcement learning techniques [14], an instance of reward-based learning, have been successfully applied in several applications. For instance, Saggar et al. [12] developed a learning algorithm for agents to optimize walks for both speed and stability in order to improve a robot's visual object recognition. The reinforcement learning approach has also been applied for designing a controller for autonomous helicopters, i.e., a helicopter learns how to hover to a place and how to fly a number of maneuvers while considering a learned policy [10]. Schneider et al. [13] introduced a reinforcement learning approach for managing a power grid.

While current work on multi-agent learning often makes use of machine learning techniques, research on agent learning from humans mostly adopts learning approaches for humans or animals. Approaches for providing agents with the ability to learn from humans can be divided into three classes: learning from advice, learning from demonstration, and learning from reinforcement (also referred to as shaping) [4]. The scenario of learning from advice is similar to learning activities of humans, where a learner gets hints from a tutor to perform an action which leads to a correct solution for a given problem. In order to be able to learn from humans, an advice can be expressed using either natural language or a scripting language. When using natural language to give advice, it is challenging to transform an advice into an understandable form for agents. Kuhlman et al. [6] created a domain-specific natural language interface for giving advice to a learner. Using a formal scripting language, coding an advice can be difficult for human trainers [9]. Both approaches, using natural language or a scripting language, are thus challenging.

Learning from demonstration, also referred to as imitation learning or apprenticeship learning, is a technique which aims at extending the capabilities of an agent without explicitly programming the new tasks or behaviors for the agent to perform. Instead, an agent learns a policy from observing demonstrations [1]. Learning from demonstration is comparable to the approach of learning from examples in an educational setting for humans. This approach can be infeasible for some tasks which require special expertise of humans to control the agent (e.g., controlling an helicopter). Taylor et al. [15] proposed a human-agent transfer (HAT) approach which combines transfer learning, learning from demonstration and reinforcement to achieve fast learning. Reinforcement learning techniques require a large amount of training data and high exploration time. Applying learning from demonstration techniques, agents learn directly from humans without explorations, and thus less time would be required compared to the reinforcement approach. However, the quality of demonstrations heavily depends on the ability of the human teacher. The work reported that combining learning from demonstration, reinforcement learning, and transfer learning to transfer knowledge from a human to an agent results in better learning performance than applying each single one.

Learning from reinforcement (or shaping) adopts the approach called clicker training for animals. In a clicker training scenario, an audible clicking device is used as a reinforcement signal when animals perform a correct action (positive assessment). The shaping approach allows a human trainer to shape an agent by reinforcing successively through both positive or negative signals. The TAMER framework is an example of the shaping approach which makes use of positive or negative assessment from humans to teach a good behavior [4]. There exist attempts for mixing the shaping approach with reinforcement learning (which provides an environmental reward within an Markov Decision Process (MDP) [14]). Isbell et al. [3] developed a social software agent which uses reinforcement learning to pro-actively take action and adapt its behavior based on multiple sources of human reinforcement to model human preferences. Knox and Stone [5] attempted to combine TAMER (which uses only human reinforcement) with autonomous learning based on a coded reward. A study showed that this combination (human reinforcement and autonomous reinforcement) leads to better performance than a TAMER agent or an reinforcement learning agent alone.

An approach based on decision tree learning which cannot be classified into the three approaches for agent-human learning above, was introduced by Thawonmas et al. [16]. The authors used a RoboCup simulation system to enable a human player to play soccer against two agents. Based on log data provided by the system, condition-action rules are derived and then applied to the agent. This way, the agent adapts decision-making behaviors of the human player. The evaluation of the system showed that the agent can show almost human decision-making behaviors in a small scenario of playing soccer after five games between a human and agents.

The learning algorithm for agents to be presented in this paper is distinct from previous work on agent-human learning in that it will deploy communication to transfer knowledge from humans to agents.

3 Case Study: Smart Airport

In order to illustrate the learning approach pursued in this paper, we use an airport departure scenario as a representative for a complex application system. The airport consists of static objects (single lanes, two-way roads, entrances, check-in counters, gates, plane parking positions, and charging stations) and moving objects (autonomous transportation vehicles (ATVs), and human-controlled vehicles (HCVs)). When there is a request of a passenger to be transported, an agency will provide vehicles (ATVs or HCVs) and manage the passenger's order. A transportation order consists of a start and an end position, pickup time, and a latest time for drop-off. The start and end positions form a route, e.g., from an entrance to a check-in counter. Since both ATVs and HCVs need energy to move, they are equipped with batteries which need to recharge regularly at charging stations. In this airport scenario, conflicts of different types can occur. We will focus on resource conflicts, i.e., two or more participants compete for one resource. Typical resource conflict situations are:

1. At least two (max. four) vehicles are approaching a crossing. One of the vehicles needs the priority to pass through the crossing first. In this situation, the resource required by the vehicles is the crossing.
2. Several vehicles are running out of energy and need to be recharged, while the charging station might be occupied. The resource required by the vehicles is the charging station.
3. Vehicles have to take passengers to unoccupied check-in counters. The resource is a check-in counter. In reality, a check-in counter usually is occupied by one or two staff members, and thus has a maximal capacity of two units.

This application scenario can be described by a multi-agent system in which autonomous vehicles are implemented by autonomous agents and human-controlled vehicles by (non-autonomous) agents. The latter can be controlled by humans, they represent (inter-)actions of humans in the system.

4 Strategy-Based Learning

Under the assumption that humans have a set of strategies for a certain conflict situation, this paper proposes a strategy-based learning approach which consists of four phases:

Phase 1: Recognizing a conflict situation. According to [17], a conflict is an opportunity for learning, because there occurs a social pressure to solve a conflict when two individuals disagree in a situation. Through resolving a conflict, individuals may change the viewpoint and their behaviors. To detect resource conflicts, a special conflict type, a conflict model and a conflict detection algorithm are required. This conflict model assumes that an agent is able to see its peers within its limited view scope. That is, each agent has information about the last, current, and next possible position of other participants existing in its scope. Thus, the conflict detection mechanism exploits the agent's *belief* about the world state within its scope. Based on this belief, an agent is able to identify other agents that will release/require a resource (an environment element) which is also required by itself. The definition for a *potential conflict* for an agent is taken from [8] as follows.

Definition 1. *Let A be an agent, its current position is $\langle X, Y\rangle$ and its next action is to require an environment element E at position $\langle X_{next}, Y_{next}\rangle$. Let α_E be the set of agents that are occupying E, $\alpha_{release,E}$ and $\alpha_{require,E}$ be the sets of agents (excluding A) that will release/require E, respectively. A has a* ***potential conflict****, denoted as $conflict(E, scope(A), \alpha_E, \alpha_{release,E}, \alpha_{require,E})$, iff $|\alpha_E| - |\alpha_{release,E}| + |\alpha_{require,E}| + 1 > C$, where C is the capacity of E and $scope(A)$ is the scope of A.*

Using Definition 1, an agent which intends to consume an environment element in the smart airport scenario, e.g., a crossing, a charging station, or a check-in counter, is able to detect potential conflicts.

Phase 2: Learning through communication. Given a conflict situation C, there exists a set of possible strategies $\{S_1, .., S_n\}$ possibly applied by a human. A strategy is defined formally as follows:

Definition 2. *A strategy is a sequence of requests and replies $\{Q_1A_1, ..., Q_nA_n\}$, where requests Q_i are carried out by a* teacher *and replies A_i are given by a* learner*. A request has one of the types: performing an action, querying data, checking a predicate, or confirming an information.*

We assume that a human has more experience than an autonomous agent, and thus a human plays the role of a *teacher* and an autonomous agent is a *learner*. The human sends requests to an agent and the agent replies to these requests. This way, the agent learns the sequence of requests which have been performed by the human in conflict situations. To establish a conversation between a human and an agent, a communication ontology is required.

In the airport departure case study, for instance, when a HCV meets an ATV at a crossing, a potential conflict occurs as described in Section 3. In this conflict situation, the human (represented by an HCV) may apply one of the following strategies: 1) calculating the priority based on the urgency of transport tasks, 2) calculating priority based on energy states of the HCV and the ATV, or 3) the strategy of politeness, i.e., give way to the participant without requesting to calculate the priority. Table 1 illustrates how the HCV initiates a conversation with the ATV when applying the task-based strategy (left column) and the politeness-based strategy (right column). For the energy-based strategy, the conversation will be similar to the task-based one, the only difference is the first request to calculate the priority based on energy states of HCV and ATV, instead. The agent distinguishes strategies applied by the human through the action requests: *calculate task priority*, *calculate energy priority*, or *give way*. As a result, the agent can "memorize" and adopt these sequences of requests in later similar conflict situations. In general, humans may use multi-modal interactions

Table 1. Conversation between human-controlled and autonomous vehicles

Task-based	Politeness-based
HCV → ATV: *calculate task priority*	HCV → ATV: I *give way*
ATV → HCV: My *priority is higher*	ATV → HCV: confirm *OK*
HCV → ATV: I *give way*	
ATV → HCV: confirm *OK*	

to indicate their strategy: e.g., using a common language, or through non-verbal acts (gestures or movements). Inferring a humans' strategy from non-verbal acts is beyond the scope of this paper. In the approach pursued in this paper, depending on the sequence of requests initiated by the human to perform actions, the agent (which has no knowledge about strategies on beforehand) can learn strategies used by humans. For instance, if the HCV sends requests to the ATV to compare the priority of two tasks, then the ATV knows that the HCV is pursuing the strategy to compare the urgency of two tasks.

Phase 3: Evaluating human's strategies. After the human has communicated with the agent, the conflict should have been solved, i.e., the resource can be allocated according to the conversation between the participants. However, participants might not be delighted with the strategy proposed by the human. For instance, an ATV might have to give way to other participants, because it has lower priority in the context of comparing transport tasks, but this ATV needs to be recharged as soon as possible because its energy state is low. Thus, this raises the need to evaluate the effectiveness of the strategy proposed by the human in each conflict situation. For this purpose, each agent involved in a conflict situation has the opportunity to rate a proposed strategy. Let the rating scalar be from the interval [0;N] where N is the best rating. The total rating for the strategy X which has been initiated by a human is $rating(X) = \sum R_A$, where

$R_A \in$ [0;N] is the rating by an agent A involved in a conflict situation. The total rating for each strategy applied by the human should be maintained and be available for all existing agents. The agents involved in the same conflict situation need to share their ratings. Here, we have to make a trade-off between a centralized coordination and intensive peer communication. Using peer communication, each agent has to send its rating to its peers. However, this solution is very communication intensive. An alternative is using a database to maintain the strategies for different conflict situations and each agent updates the total rating for each strategy. In the approach followed in this paper, we choose the second option.

Phase 4: Applying the best strategy. When an autonomous agent detects a conflict with other agents in a situation in which it had a conflict with humans before, the agent takes the set of strategies which have been collected by learning from humans to compare. The best strategy of this set is determined by querying the rating in the strategy database. The strategy which has the highest rating is taken as the best one which can be used. Once again, after the agent has applied that strategy, its peers have the possibility to update the total rating in the strategy database. A question is which agent should initiate a conversation in case of a conflict situation where no human is involved. For this purpose, either an agent who has acquired knowledge should initiate the conversation or one of the involving agents is selected randomly.

5 Implementation

We implemented the airport scenario using the JRep simulation platform [2]. JRep is an integration of Repast Symphony and the JADE Framework. Repast provides a toolkit for visual simulations of multi-agent systems. JADE supports developing intelligent behaviors for agents and provides communication protocols according to FIPA-ACL[1]. In order to enable a conversation between agents and a human, we need to define a communication ontology. We extended JRep with the possibility to add human-controlled vehicles (HCVs) which can be controlled via the four arrow keys of the keyboard. Note, if there exist several HCVs in the airport setting, only one HCV can be moved at a time (while other HCVs cannot perform any action). This is a limitation of our current simulation framework.

The left hand side of Figure 1 illustrates the simulation of a conflict situation at a crossing in a smart airport. The right hand side shows the existence of current agents in the system and the agent management GUI (e.g., to control the communication between agents) provided by JADE.

6 Evaluation

The goal of the evaluation study is to prove that the strategy-based learning algorithm is beneficial for multi-agent systems. For this purpose, we test the two hypotheses mentioned in Section 1.

[1] http://www.fipa.org/specs/fipa00061

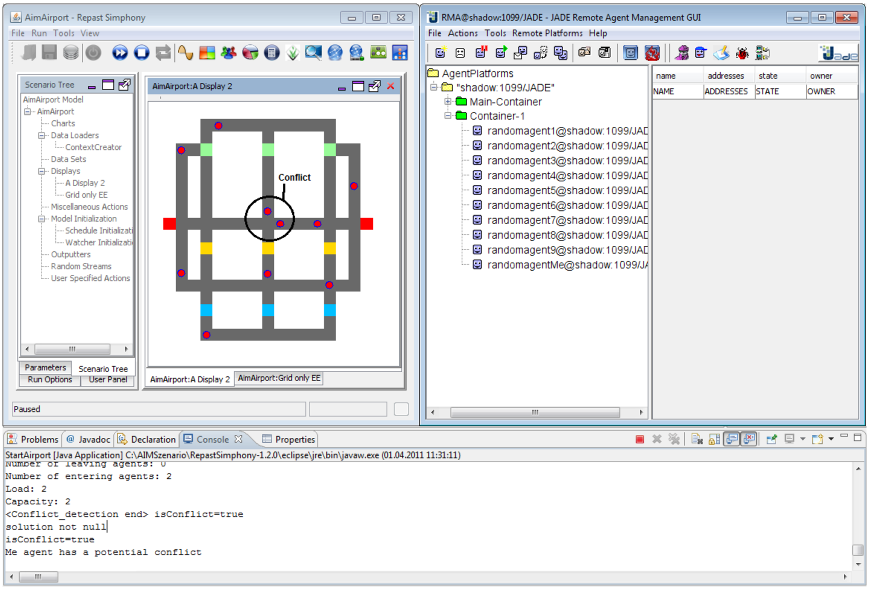


Fig. 1. Enhanced Simulation Platform

6.1 Design

To carry out the evaluation study, we used the simulation framework described in the previous section. In the first round, we carried out six simulations, each with 30 ATVs. The number of HCVs was varied between zero and five. For each simulation, the system was run with 100 ticks. Each tick represents a movement step of an agent or a conversation (consisting of several message exchanges) between an HCV and an ATV (or between ATVs). Then, in the second round, similarly, six other simulations were executed, each with 50 ATVs (and with zero to five HCVs). Results of the second round were used to verify the outcomes of the first round. Each ATV can move randomly around the airport map. If an ATV detects a resource conflict on the road, i.e., the road cell capacity is reached, then the ATV chooses another movement direction. In our study, autonomous agents (ATVs) that have interacted with humans (HCVs) in conflict situations and have thus learned problem solving strategies were able to apply those strategies in the similar conflict situation. Agents without any strategy for solving a conflict situation did not survive conflicts.

6.2 Results

Table 2 shows statistical results of six simulations with 50 ATVs and six simulations with 30 ATVs. The third and the fifth columns of the table show the number of interactions between ATVs and HCVs during 100 simulation ticks.

Table 2. Results of interactions between agents and humans

HCVs	With 50 ATVs		With 30 ATVs	
	Survived ATVs	**Interactions**	**Survived ATVs**	**Interactions**
0	16	0	7	0
1	17	9	17	21
2	30	56	19	42
3	25	47	20	52
4	24	57	20	60
5	29	54	24	57
		m=44.6 (s.d.=20.3)		m=46.4 (s.d.=15.8)
	Spearman's ρ=0.66		Spearman's ρ=0.9	

As mentioned in Section 5, during one tick, only one human could interact with agents. In average, we simulated between 44 and 46 ATV-HCV interactions. In order to test whether a higher amount of ATV-HCV interactions resulted in a higher stability of the system, we computed the correlation between the number of ATV-HCV interactions and the number of survived ATVs using the statistics software provided by the Office for Research Development and Education[2]. From the table, we can notice that, in general, through interactions with humans, the number of survived ATVs is higher than in simulations without the existence of humans (indicated by the first row). The Spearman's coefficient shows that for simulations with 30 ATVs, a strong correlation between the number of ATV-HCV interactions and the number of survived ATVS (ρ=0.9) can be identified. For simulations with 50 ATVs, the correlation coefficient (ρ=0.66) is lower, but still relatively high, and shows a moderate positive correlation between the number of ATV-HCV interactions and the number of survived ATVs. As a conclusion, the hypothesis that the more opportunities agents learn problem solving strategies from humans, the less agents will die, can be confirmed.

Table 3 shows how agents adapted their bahavior through interactions with humans. The left part and the right part of the table present the relative frequencies of the strategies applied by HCVs and ATVs, respectively. In boldface, the most frequently used strategy is highlighted. During the simulations with 30 ATVs, we can recognize that the strategy adopted by most agents is always consistent with the strategy chosen most frequently by HCVs, although the relative frequencies differ (ATVs tend to focus on one primary strategy, while HCVs exhibited a more varied behavior). For the simulation with 50 ATVs, the table shows the same tendency. One exception: in the case of the simulation with the existence of 5 HCVs, it is not clear which strategy (politeness or task-based) was favored by HCVs, while the ATVs have decided for the task-based strategy. Hence, the hypothesis that as a result of applying the strategy-based learning algorithm, agents will adopt the strategy most humans applied, can be confirmed.

[2] Wessa, P. (2011), Free Statistics Software, version 1.1.23-r7, URL `http://www.wessa.net/`

Table 3. Adaptation of agents

HCVs	Strategy applied by HCV (%)			Strategy applied by ATV (%)		
	Politeness	Task-based	Energy-based	Politeness	Task-based	Energy-based
Simulations with 30 ATVs						
1	**100**	0	0	**100**	0	0
2	42.9	**57.1**	0	2.4	**97.6**	0
3	**40.4**	30.8	28.8	**97.9**	0	2.1
4	**46.7**	33.3	20	**77.8**	22.2	0
5	**57.9**	29.8	12.3	**96.5**	3.5	0
Simulations with 50 ATVs						
1	**100**	0	0	**100**	0	0
2	33.9	**66.1**	0	6.5	**93.5**	0
3	27.7	**38.3**	34	21.7	**72.2**	6.1
4	**43.9**	14	42.1	**73.9**	0	26.1
5	**40.7**	**40.7**	18.5	0	**100**	0

7 Conclusion

In this paper, we have presented a strategy-based learning algorithm for agents through communication with humans. The learning process consists of four phases: 1) detecting conflict situations, 2) humans initiating a conversation with agents and deciding on a conflict solving strategy, 3) agents involved in the conflict situation rate the effectiveness of the proposed strategy, and 4) the agent applies the most effectively rated strategy in a similar situation. We have conducted a pilot study to evaluate the benefits of this learning algorithm. The evaluation shows that the multi-agent system with interactions between humans and agents becomes more stable than a system without interactions with humans, and that agents adopt the problem solving strategy applied most frequently by humans. Note, the evaluation assumed that the agents have no knowledge how to solve conflicts as long as they have not interacted with humans. This learning approach for agents is novel in that it exploits the communication ability of agents (using the FIPA-ACL protocols) to be instructed by humans, whereas most existing work is based on machine learning techniques. In the future, we will try to shorten the conversation steps between humans and agents, because a long conversation is not appropriate for traffic situations and is resource expensive.

References

1. Argall, B.D., Chernova, S., Veloso, M., Browning, B.: A Survey of Robot Learning From Demonstration. Robotics and Autonomous Systems 57(5), 469–483 (2009)
2. Görmer, J., Homoceanu, G., Mumme, C., Huhn, M., Müller, J.P.: JRep: Extending Repast Simphony for Jade Agent Behavior Components. In: Proceedings of the IEEE/WIC/ACM Int. Conf. on Intelligent Agent Technology, pp. 149–154 (2011)
3. Isbell, C., Kearns, M., Singh, S., Shelton, C., Stone, P., Kormann, D.: Cobot in LambdaMOO: A Social Statistics Agent. Autonomous Agents and Multiagent Systems 13(3), 327–354 (2006)

4. Knox, W.B., Stone, P.: Interactively Shaping Agents via Human Reinforcement - The TAMER Framework. In: Proceedings of the 15th International Conference on Knowledge Capture, pp. 9–16. ACM, New York (2009)
5. Knox, W.B., Stone, P.: Combining manual feedback with subsequent MDP reward signals for reinforcement learning. In: Proceedings of the 9th Int. Conference on Autonomous Agents and Multiagent Systems, vol. 1, pp. 5–12. AAMAS (2010)
6. Kuhlmann, G., Stone, P., Mooney, R.J., Shavlik, J.W.: Guiding a Reinforcement Learner With Natural Language Advice: Initial Results in RoboCup Soccer. In: Proceedings of the AAAI Workshop on Supervisory Control of Learning and Adaptive Systems (2004)
7. Le, N.T., Menzel, W., Pinkwart, N.: Considering Ill-definedness of Problems From The Aspect of Solution Space. In: Proceedings of the 23rd International Florida Artificial Intelligence Conference (FLAIRS), pp. 534–535. AAAI Press (2010)
8. Le, N.T., Märtin, L., Pinkwart, N.: Learning Capabilities of Agents in Social Systems. In: Proceedings of The 1st International Workshop on Issues and Challenges in Social Computing (WICSOC), held at the IEEE International Conference on Information Reuse and Integration (IRI), pp. 539–544. IEEE, NJ (2011)
9. Moreno, D.L., Regueiro, C.V., Iglesias, R., Barro, S.: Using Prior Knowledge to Improve Reinforcement Learning in Mobile Robotics. In: Proceedings of Towards Autonomous Robotic Systems (TAROS), Technical Report Series, Report Number CSM-415, Department of Computer Science, University of Essex (2004)
10. Ng, A.Y., Kim, H.J., Jordan, M.I., Sastry, S.: Inverted Autonomous Helicopter Flight Via Reinforcement Learning. In: International Symposium on Experimental Robotics. MIT Press (2004)
11. Panait, L., Luke, S.: Cooperative Multi-agent Learning: The State of the Art. Autonomous Agents and Multi-Agent Systems 11(3), 387–434 (2005)
12. Saggar, M., D'Silva, T., Kohl, N., Stone, P.: Autonomous Learning of Stable Quadruped Locomotion. In: Lakemeyer, G., Sklar, E., Sorrenti, D.G., Takahashi, T. (eds.) RoboCup 2006: Robot Soccer World Cup X. LNCS (LNAI), vol. 4434, pp. 98–109. Springer, Heidelberg (2007)
13. Schneider, J., Wong, W.K., Moore, A., Riedmiller, M.: Distributed Value Functions. In: Proceedings of the 16th International Conference on Machine Learning, pp. 371–378. Morgan Kaufmann (1999)
14. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
15. Taylor, M.E., Suay, H.B., Chernova, S.: Integrating Reinforcement Learning with Human Demonstrations of Varying Ability. In: Proceedings of the 10th Int. Conference on Autonomous Agents and Multiagent Systems, pp. 617–624. AAMAS (2011)
16. Thawonmas, R., Hirayama, J.-I., Takeda, F.: Learning from Human Decision-Making Behaviors - An Application to RoboCup Software Agents. In: Hendtlass, T., Ali, M. (eds.) IEA/AIE 2002. LNCS (LNAI), vol. 2358, pp. 136–145. Springer, Heidelberg (2002)
17. Weiß, G., Dillenbourg, P.: What is 'multi' in Multi-agent Learning. In: Dillenbourg (ed.) Collaborative-learning: Cognitive, pp. 64–80. Pergamon Press, Oxford (1999)

Ontology Based Knowledge Management and Learning in Multi-Agent System

Dariusz Choinski and Michal Senik

Silesian University of Technology, ul. Akademicka 16, 44-100 Gliwice, Poland
{dariusz.choinski,michal.senik}@polsl.pl

Abstract. This paper presents implementation of an algorithm that handles knowledge management and learning processes which are implemented in the existing Multi-Agent System (MAS). Both presented knowledge management and learning mechanisms are based on the implemented ontology mechanisms. Proposed MAS solution is based on the Java Agent Development (JADE) framework which is a Java based tool that supports creation of FIPA (Foundation for Intelligent Physical Agents) compliant agents. It integrates with various different OPC (OLE for Process Control) and database servers simultaneously. OPC is an open automation communication interface that is used to retrieve and analysis of various different real time data from the various different remote process controllers. To integrate OPC automation interface, presented MAS utilize Java Native Interface (JNI) which is a bridge between a Java and native programming languages. Presented MAS establishes database cooperation by means of the NHibernate entity framework which is a .Net platform specific, object relational mapper (ORM). Integration with .Net platform is performed by means of the highly refactored JADE Sharp module add-on which enables creation .Net agents compliant with the JADE framework.

Keywords: Multi-Agent Systems, Java, .Net, JADE, FIPA, OPC, XML, NHibernate, hybrid systems, concurrent programming, knowledge sharing, learning.

1 Introduction

In the agent managed environment reasoning capabilities are far more sophisticated than in the traditional object oriented solutions because in the MAS reasoning process is strongly supported by means of ontology [1], [2]. Conceptually ontology itself is a static description of the surrounding, dynamic environment. It is a formal explicit hierarchical representation of the knowledge about the specific domain based on which MAS will be created and because of that it should be the first element created of each MAS. It is worth mentioning that MAS can utilize various numbers of ontologies which means that it is by default open for the initially unknown modifications or other functional elements such as new agents. Fundamental feature of each MAS is the aspect of ontology based knowledge sharing. In most cases ontology is considered as system knowledge. This knowledge is modelled and

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 65–74, 2012.

implemented beforehand MAS application and it is known prior to its runtime [3], [4]. Ontology not only describes static properties of the system but also runtime states in which the system can reside as well as the conditions of transitions between those states. Based on the ontology, MAS agents can share their knowledge and work together i.e. cooperate over the integrated system's problems because knowledge sharing is the natural way for single pieces of MAS to achieve their goals. Technically ontology is a set of simple and more complex rules such as concepts, predicates and actions which can be created in many available programming environments such as Java or .Net. Additionally ontology can be created in the XML language which can be considered as a far more convenient way in comparison to the Java or .Net based solutions because of the fact that XML based ontology can be reused in each mentioned programming environment. Each ontological expression can be organized in a hierarchical structure which means that simpler entities can be nested in more complex entities. Agent's reasoning capabilities reuse those ontological structures during concurrent cooperation processes over many different integrated system parts which assure high quality and accuracy of the retrieved data. Such approach allows for dynamic management of system knowledge using its static description.

2 Knowledge Ontology Based Memory Management

In many situations information is passed between sender and receiver without any prior knowledge about what is the current state of the receiver especially what is the current state of the receiver knowledge. In case of rather fast data collection the communication process itself might become inefficient. This is because of the fact that large amount of the raw static data will be passed continuously between the sender and receiver unnecessarily increasing the communication workload and message processing time. Memory management concept however enables various pieces of the integrated system to optimize and in effect reduce their need for communication (fig. 1).

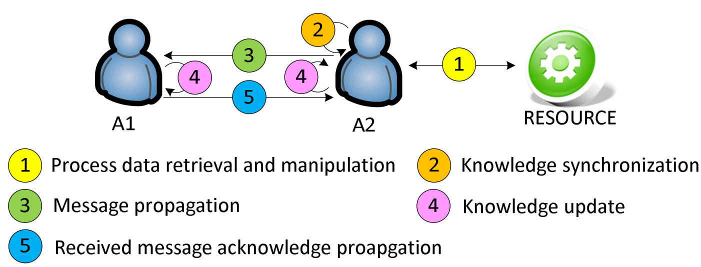


Fig. 1. Memory management mechanism

One approach to solve this problem would be to programmatically implement dedicated memory management mechanism and integrate it with an already existing system. Introduced mechanism must be flexible enough to handle various unexpected scenarios in the integrated system. Implementation of such memory management mechanism ought to be resistant to the possibility of dynamic integrated system modifications. Moreover in case of considerably larger structural modifications introduced mechanism must be flexible enough to be modified i.e.: extended during maintenance implementation. This means that the memory management mechanism must be scalable enough to follow the extension of the integrated system which means as well that the mechanism must present an open architecture. Generally, implemented memory management mechanism ought to be a factor that guards system's data integration quality and optimizes communication process. It is worth mentioning that implemented memory management mechanism might become quite useful whenever decision has to be made about from which point of the integrated system the most actual data could be obtained and processed. This is because implement memory management mechanism assures that the same quality data can be obtained faster from the other parts of the system rather than from the original source of the data. Such tunneled approach is literally more convenient way of accessing the data because there is no need anymore to reach the source of the data in order to process it. Generally, knowledge management is as simple as comparison of two different ontological concepts goes but beyond that it is a very complex and distributed process. Knowledge sharing and management process (fig. 1) is always a two side task that occurs always whenever two different agent entities enter into the cooperative mode. Cooperation is the normal way in which two different agents share their knowledge over a common problem via a set of a various different preconfigured messages types. Each message carries an ontological expression that can be reused by the receiver agent to not only perform certain tasks but to update its current knowledge state as well. Agent knowledge ought to be updated whenever an agent realizes that passed ontology expression modifies its current state. As a response receiver agent ought to generate acknowledge message containing original message content. This message should traverse back to the sender agent modifying its knowledge about the receiver agent. Each time sender agent sends new message to the receiver agent, message ontological content is compared against the knowledge sender agent gathered over time on the receiver agent. This allows sender agent to send only the needed data. It also drastically minimizes network traffic workload and agent message processing time.

Presented approach has been widely adopted by the each platform agent. Knowledge ontology based memory management mechanism is build up dynamically according to the current state of the integrated system. Its design established at the beginning of the implementation stage was dedicated to handle unpredicted but naturally possible, future agent platform modifications. Implemented knowledge management mechanism is closely coupled with both JADE platform event and subscription based messages propagation mechanisms. Platform events such as agent creation or deletion are essential to the knowledge ontology based memory management mechanism. Subscription and event based approach to messaging process are similar at a glance but differs when it comes to the implementation. Event based messaging mechanism is a feature of each JADE agent because AMS [2] agent status messages are propagated to each available platform agent. Subscription based messaging mechanism is an additional feature that can be implemented by an agent but it is not required. Subscription based mechanism is based on the JADE DF agent discovery mechanism and AMS [2] agent events propagation. Each time an agent wants to share services it provides it must register itself along with services it provides with the JADE DF agent. Whenever new agent is born creation notification event message is propagated to each single platform agent thus enabling cooperating agents for the newborn agent memory object creation. Each time existing agent dies deletion notification event message is propagated to the each single platform agent causing removal of the proper agent memory object slot from the each cooperating platform agent memory objects collection. This is conceptually the fastest and the most convenient way in which each agent can either manifest its presence in the agent platform so it can be discovered and update its knowledge about the current state of the whole platform. Subscription driven messaging propagation mechanism is the second basic concept that enables platform agents to dynamically build up their knowledge. It allows obtaining all needed pieces of information without any prior request. This is also the most convenient way in which any new born agent builds up its knowledge about the existing agent platform.

Technically, agent memory is represented by a shared public abstract class that defines common set of agent specific properties and presents them via a publicly visible set of methods. Each new agent that implements memory management mechanism simply extends this class and performs customization individually. Customization of the agent memory subclass is usually based on addition of the new ontological field types. Agent memory subclass is used during runtime to instantiate an agent memory object whenever agent creation notification message arrives. This object is then put into the dynamically maintained agent memory object collection. This collection always holds the exact number of memory objects as the number of the cooperating agents. Each agent updates its memory object collection over its lifetime which assures freshness of the acquired data. This approach enables agent to become an additional source of the processing data that can be accessed instead of reaching back to the original source of the data. Each agent memory object corresponds to the cooperating agent and contains a constant number of fields that can be accessed via a public API. Each agent memory field can be represented by a single-valued or collection-valued reference variable of simple, non-ontological or complex, ontological expression type. It is worth mentioning that memory management relates only to the ontology concept type which is the most basic memory element. Ontology concept type characterizes the dynamic properties of the tiniest parts of the integrated system and when combined together with other concept types, forms a complete, hierarchical description of the greater system structure. Each presented solution specific ontology concept inherits from the main, top level concept class that defines a few publicly visible methods that are used during agent runtime to synchronize, update or delete its memory.

3 MAXS Platform

MAXS (Multi Agent Cross Platform System) platform is a real time processing, hierarchical, multilayered, MAS capable of dynamical adjustment to the existing information structure [5], [6]. It wraps over existing JADE framework [2] using a set of newly implemented agents thus creating a complex system. Its existence is dictated by today's constantly arising problem of having rapidly changing environment, which should be handled with extreme caution in order to maintain high level of quality of acquired data. MAXS platform is designed especially to be responsible of dealing with a situation of having probability of such dynamic changes. It can function on one or more remotely controlled hosts, which may also vary in time. This indicates that MAXS platform can adapt to almost any existing situation. In the current development stage, MAXS platform can interact with various different OPC DA [7] and ordinary database servers in order to obtain, store and process real time data. In addition to the JADE FIPA [2] specific communication process, MAXS platform agents simultaneously reuse integrated databases as a fully functional, fully qualified, parallel communication channel. Such redundancy can be very helpful whenever one channel fails unexpectedly communication process can still occur by means of the secondary channel without having the problem of either data integrity or performance degeneration. Normally, MAXS reuses database communication channel to store or update both raw process data obtained from the system and platform configuration settings that can be reused during the next system startup. Integration with various database servers is achieved by means of the NHibernate [8] entity framework which is a popular, mature, open source, .Net platform specific object relational mapper (ORM) which is based on Java's Hibernate project. ORM as a programming technique allows for the conversion between various incompatible type systems, creating as a result a virtual object database that can be accessed from the object oriented programming languages such as for example Java or C-sharp. It simply translates between the database's relational model of tables, columns, and keys to the application's object model of classes and properties. NHibernate framework however, is designed to be used over the .Net platform only and in order to utilize it MAXS platform was additionally integrated with the .Net platform by means of the JADE Sharp. JADE Sharp is a JADE add-on which comes as an additional .dll module which enables creation .Net agents compliant with the JADE framework. Original version of the JADE Sharp was proposed by the TILab which has also created the JADE framework itself. However, for the purpose of the proposed MAS, JADE Sharp add-on module was strongly modified and refactored to fit the needs of the MAXS platform. By means of the modified JADE Sharp module MAXS platform can integrate not only with the Java platform but with the .Net platform as well. In order to establish efficient cross-platform communication each MAXS agent reuses common XML based messaging mechanism. The MAXS platform establishes efficient cooperation with various different OPC Servers through the Java Native Interface (JNI) [9] which bridges Java mechanisms into the native language world such as C/C++, Delphi or through the OPC Foundation .Net automation wrapper library. Each solution present the same functionalities, however each one is different in the approach as the first solution is strictly Java based whilst the second one is based on Microsoft .Net framework.

OPC is an open automation interface protocol for fast data retrieval and analysis and it is being actively developed by OPC Foundation since 1994. Until now OPC Foundation introduced a set of different communication standards amongst which the most widely used is OPC Data Access (OPC DA) interface which is used for retrieval of various real time data from remote data sources (remote process controllers) [7]. It provides an easy, secure and generic way in which each client application can connect with various types of OPC DA Servers which basically are a third party pieces of software that provide OPC protocol specific data from various types of controllers. Moreover, OPC DA Server can be accessed by a various number of OPC Client applications simultaneously, providing them with the needed information.

MAXS platform in the current development stage consists of nine different types of agents: Supervisory Agent (SA), Node Agent (NA), Port Scanner Agent (PSA), Network Scanner Agent (NSA), Management Agent (MA), OPC Agent (OA), Discovery Agent (DA), Lifecycle Supervisory Agent (LSA) and Database Agent (DBA). Layered MAXS architecture [10] concept is visible on the fig. 2. In short SA is used to dynamically administer other platform agents that show up in the platform during runtime. Its functionality is strictly high level, which means that it can terminate, suspend or create any platform agent. It can handle the process of agent relocation as well. NSA is responsible for establishing a network structure. It is designed to discover available remote host locations. Information about each accessible network node is transferred back to the SA for further processing. PSA is responsible for port scanning processes. It discovers enabled ports on the remote machine and transfers this information directly to the SA for further processing. DA is responsible for remote host environment discoveries. It acquires the data about available OPC Server services and creates the exact OA pool as the number of discovered OPC Servers. Each OA is then linked by DA to the single one OPC Server. OA is responsible for processing OPC Server data. It can configure OPC Server with OPC Groups and OPC Items. It can reconnect locally or remotely between various types of different OPC Servers but it interacts with only one. As a result OA propagates asynchronously gathered OPC Server specific data to all available listeners. LSA is closely coupled with OA because of the possible instability of the OA connection with third party OPC Server services. It synchronously checks whether assigned OA is active and if not LSA restores assigned OA to life based on the latest configuration. DBA is a JADE Sharp agent that is used to configure and maintain connection with the database. When active it stores or retrieves MAXS platform specific i.e. OPC Server or JADE specific data. MA is responsible for user interaction with various different OA agents. It is an agent with GUI which allows user for real time data manipulation. The number of MA agents varies in time because it is strictly related to the number of users operating in the system. MA is designed to receive notifications from all available OA agents. Each MA can receive the same data concurrently by means of its DBA agent. NA is responsible for manual remote host registration to the existing MAXS platform. Complete description of platform architecture and each platform agent was presented in [6] and [10]. In addition to those agents there are two more agents that are JADE specific entities. Those agents are DF (Directory Facilitator) and AMS (Agent Management System). The DF agent is an optional component that starts whenever JADE starts whilst AMS presence is required for any JADE platform to work properly. Complete overview of the DF and AMS agents can be found in [2]. MAXS architecture presents openness and flexibility which means that new agents can be activated or deactivated dynamically according to the current environment needs which can vary over time. This implies self management nature of the presented platform which is a key concept that satisfies changing conditions of the integrated environment. Self management functionality enables MAXS platform agents to address not only strictly integrated environmental issues but MAXS platform specific events such as unexpected agent's internal errors or possible instabilities as well. Such approach allows fast situation diagnosis and proper MAXS counteraction. MAXS self management ability is a fundamental feature that guards and preserves system data integrity and quality.

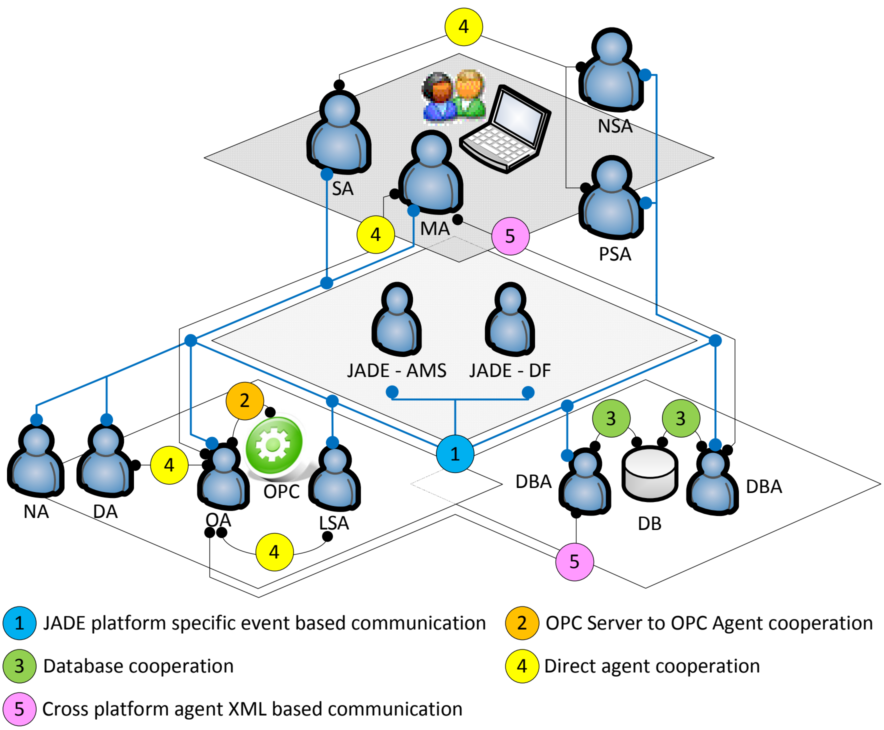


Fig. 2. MAXS architecture

Each MAXS agent shares similar capabilities making them more compliant and consistent. Those capabilities are the key concepts of the internal platform architecture and involve mobility, cloning, common messages, behaviors, threading, ontology, event subscription, message notifications and message propagation mechanisms [6]. To complete assigned tasks MAXS agents reuses a common set of fully customized and concurrent behaviors. Each behavior extends common mechanism of knowledge sharing and management that utilizes MAXS platform ontology. MAXS reasoning model is based on the subscription and notification mechanisms as well as on the agent knowledge which is gathered over a given period of time from other platform agents. That way each interacting agent can reason about the current needs of the receiver set and exchange only necessary data which minimizes network workload and improves agent performance. MAXS platform ontology bridges OPC DA interface protocol specification into MAXS internal mechanisms. Moreover, it links this information into higher level ontological structures which can be analyzed by each single MAXS agent. It has been created to not only describe static system characteristics but also to be used whenever OPC DA interface protocol native asynchronous readouts are performed i.e. server status, tags properties, tags values triplets (value, quality and timestamp) concurrently. By means of the MAXS platform ontology each platform agent is able to share its current knowledge state with the other MAXS platform listeners, which is a natural agent-based knowledge management decomposition problem. Moreover, ontology is extensively used during system knowledge synchronization and update processes. This commonly used by the each MAXS agent concept is extremely important during extensive messaging processes.

4 MAXS Learning Mechanism

Each MAXS agent memory object is updated during agent cooperation process which is based on the rich set of various kinds of different, preconfigured agent behaviors. Because of the fact that MAXS platform handles process of fast data collection from the various different OPC Servers, agent memory objects are used to support this process in the upper layers of the agent platform. In order to cooperate with OPC Servers efficiently MAXS platform reuses dedicated to Java JNI mechanism. JNI is a fast way in which OPC Server specific data can be processed inside Java programming environment. It is fast also because of the short physical distance between particular OPC Server and cooperating MAXS agent. In case of a larger OPC Server data structures, communication process with the other layers of the integrated system might become a very time consuming, inefficient thus slow process. It is worth mentioning that the same situation occurs whenever more than just one OPC Server exists in the system. However, the problem of fast data collection in the upper layers of the MAXS platform does not exist. This is mainly because of the availability of the agent memory management and learning mechanisms which are used to determine whether particular information needs to be propagated or not. Fig. 3 presents a concept which is used in the MAXS platform to establish agent knowledge i.e. learn an agent about not only the ongoing process itself but about cooperating agent current state as well. MAXS platform presents an asynchronous, dynamic and continuous knowledge learning model. This means that some parts of the knowledge are constantly updated whenever a change in the source of the data occurs. Dynamic of the described learning mechanism is presented on the Petri net visible on the fig.3. MAXS knowledge learning model assumes that agent updates its knowledge each time a new message is received but in order to assure the sender that the message is properly understood, receiver agent returns the same message to the sender agent. Sender agent in response updates its knowledge about the receiver agent and whenever a change in the source of the data occurs comparison of the actual state of the receiver agent and the actual state of the data source will be performed. This can result with a new notification message that will be propagated to the receiver agent but this time the message will be synchronized. Synchronized message is composed of only those pieces of data that needs to be updated. It is because the data that is the same at both sides will be cut of from the message. It is worth mentioning that most of the OPC Server specific data is rather static so there is no need to repeatedly propagate everything that OPC Server produces. Similar situation occurs whenever an additional, intermediate source of the raw process data is available. In the case of the MAXS platform a database server stands as a natural example of such indirect source. MAXS platform treats database server equally to the JADE FIPA specific communication channel. This means that the same data can be obtained in two completely disjoint ways, faster and without any performance hit. Database can be treated as huge, natural, failsafe agent memory object that can endure most of the integrated system instabilities or other unpredicted failures. Fig. 4 conceptually presents how such database integration looks in the MAXS platform

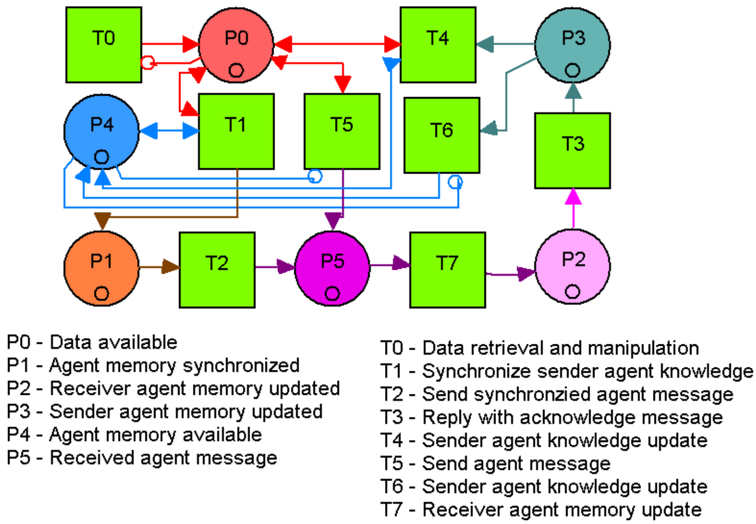


Fig. 3. MAXS knowledge learning algorithm **– Petri net diagram**

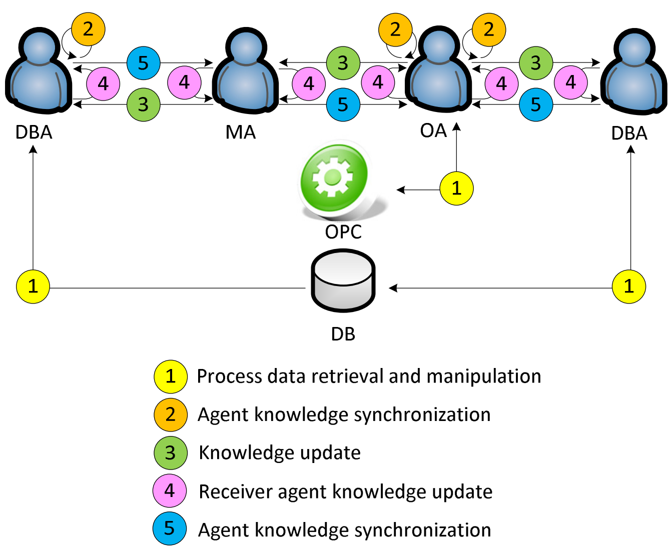


Fig. 4. MAXS heterogeneous knowledge learning algorithm

5 Conclusions

Presented hybrid, ontology based knowledge management and learning mechanisms are based on the both MAS and database server communication channels. Such approach assures and preserves data integrity, accuracy and quality in the instable and dynamic environment. Both presented mechanisms significantly reduce the amount of communication workload. It is because the state of the receiver agent is continuously monitored, updated and synchronized with the source of the data by the sender agent. Introduced mechanisms can withstand each unpredicted integrated system disturbance such as asynchronous user system cooperation. This is because user may need to perform an unexpected system online or offline data modification. It is worth mentioning that the online data modifications are performed during integrated system runtime which can throw whole system from one state to the completely new one [11]. However, the self management nature of the presented MAS can preserve stability of the integrated system by adjusting itself automatically to the new conditions. It is because of the fact that asynchronous modifications are in fact expected to occur in the MAS based systems.

Acknowledgments. This work was partially supported by the Polish Ministry of Scientific Research and Higher Education N N514 471539.

References

1. Wooldridge, M.: An Introduction to Multiagent Systems. John Wiley & Sons Ltd. (2002)
2. Bellifemine, F., Caire, G., Greenwood, D.: Developing multi-agent systems with JADE. John Wiley & Sons, Chichester (2007)
3. Choiński, D., Metzger, M., Nocoń, W.: Multiscale Three-Phase Flow Simulation Dedicated to Model Based Control. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part II. LNCS, vol. 5102, pp. 261–270. Springer, Heidelberg (2008)
4. Metzger, M.: Comparison of the RK4M4 RK4LIN and RK4M1 methods for systems with time-delays. Simulation 52, 189–193 (1989)
5. Leduc, R.J., Lawford, M., Dai, P.: Hierarchical Interface-Based Supervisory Control of a Flexible Manufacturing System. IEEE Transactions on Control Systems Technology 14, 654–668 (2006)
6. Choinski, D., Senik, M.: Multi-Agent Oriented Integration in Distributed Control System. In: O'Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS (LNAI), vol. 6682, pp. 231–240. Springer, Heidelberg (2011)
7. Iwanitz, F., Lange, J.: OPC – Fundamentals, Implementation and Application. Huthig Verlag, Heidelberg (2006)
8. Dentler, J.: NHibernate 3.0 Cookbook. Packt Publishing Ltd., Birmingham (2010)
9. Liang, S.: The Java Native Interface Programmer's Guide and Specification. Addison-Wesley (1999)
10. Choinski, D., Senik, M.: Multilayer Automated Methods for the System Integration. In: Luo, Y. (ed.) CDVE 2011. LNCS, vol. 6874, pp. 86–93. Springer, Heidelberg (2011)
11. Metzger, M., Polakow, G.: A Survey on Applications of Agent Technology in Industrial Process Control. IEEE T. Ind. Inform. 7(4), 570–581 (2011)

Data Analysis Systems and Cognitive Modelling Processes

Lidia Ogiela and Marek R. Ogiela

AGH University of Science and Technology
Al. Mickiewicza 30, PL-30-059 Krakow, Poland
{logiela,mogiela}@agh.edu.pl

Abstract. This publication proposes using biological modelling methods based on DNA cryptography to semantically analyse data. The solutions proposed are illustrated with the example of E-UBIAS systems which analyse image data in combination with the identity analysis (personalisation of individuals). The use of DNA cryptography to analyse data makes it possible to unanimously assign analysed data to an individual at the personal identification stage. At the same time, supplementing the system with semantic analysis processes conducted based on semantic interpretation allows the possible lesions that the person suffers from to be identified.

Keywords: DNA cryptography, E-UBIAS systems (Extended Understanding Based Image Analysis Systems), semantic analysis, personalisation and personal identification processes.

1 Introduction

More and more research on the semantic analysis of image data is carried out nowadays. Such research papers deal with the design of UBIAS systems (Understanding Based Image Analysis Systems). The essence of the presented class of systems consists in the semantic analysis of images being interpreted, and particularly of medical images portraying lesions of different organs [4-12]. Systems that analyse hand bones represent a special class of UBIAS systems because this class of systems can be used not only to semantically analyse hand bone lesions, but have become the basis for designing a new class of systems: E-UBIAS. E-UBIAS systems (Extended Understanding Based Image Analysis Systems) carry out personalisation and identification analyses (Fig. 1).

E-UBIAS systems carry out personal verification and identification processes with the addition of hand shape image analysis and hand lesion detection. These systems have been developed as a result of work conducted on analysing images of hand bones in UBIAS systems [4-12].

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 75–83, 2012.

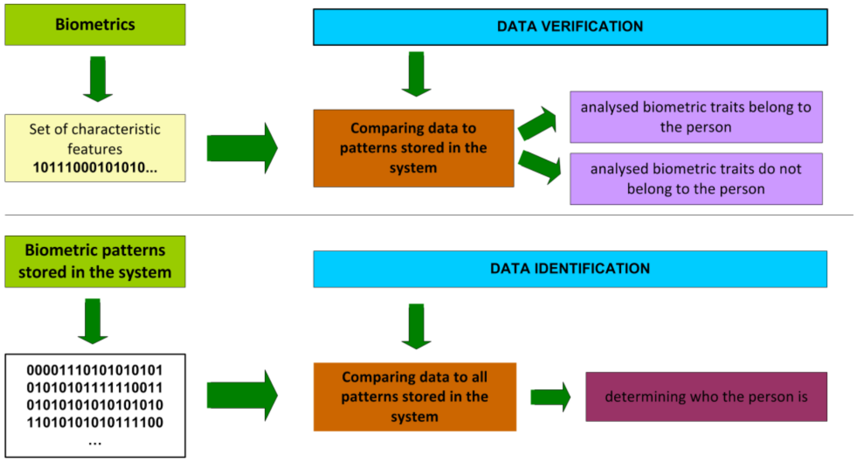


Fig. 1. Personal verification and identification processes

2 Hand Bone Analysis in E-UBIAS Systems

Systems that semantically analyse images showing hand bone lesions have been described in multiple scientific publications, particularly in [10-12].

The operation of these systems is essentially based on the use of linguistic formalisms, namely a defined graph grammar:

$$G_{hand} = (N_{hand}, T_{hand}, \Gamma_{hand}, S_{hand}, P_{hand})$$

where:

N_{hand} – denotes set of non-terminal symbols:

N_{hand} = {ST, ULNA, OS SCAPHOIDEUM, OS LUNATUM, OS TRIQUETRUM, OS PISIFORME, OS TRAPEZIUM, OS TRAPEZOIDEUM, OS CAPITATUM, OS HAMATUM, m1, m2, m3, m4, m5, SES1, SES2, PIP1, PIP2, PIP3, PIP4, PIP5, PM2, PM3, PM4, PM5, PD1, PD2, PD3, PD4, PD5}

T_{hand} – denotes set of terminal symbols:

T_{hand} = {r, u, s, l, t, p, tm, tz, c, h, m1, m2, m3, m4, m5, s1, s2, pip1, pip2, pip3, pip4, pip5, pm2, pm3, pm4, pm5, pd1, pd2, pd3, pd4, pd5}

Γ_{hand} – {p, q, r, s, t, u, v, w, x, y, z} – the graph defining relations between hand bones (Fig. 2)

S_{hand} – the start symbol

P_{hand} – set of productions (Fig. 2).

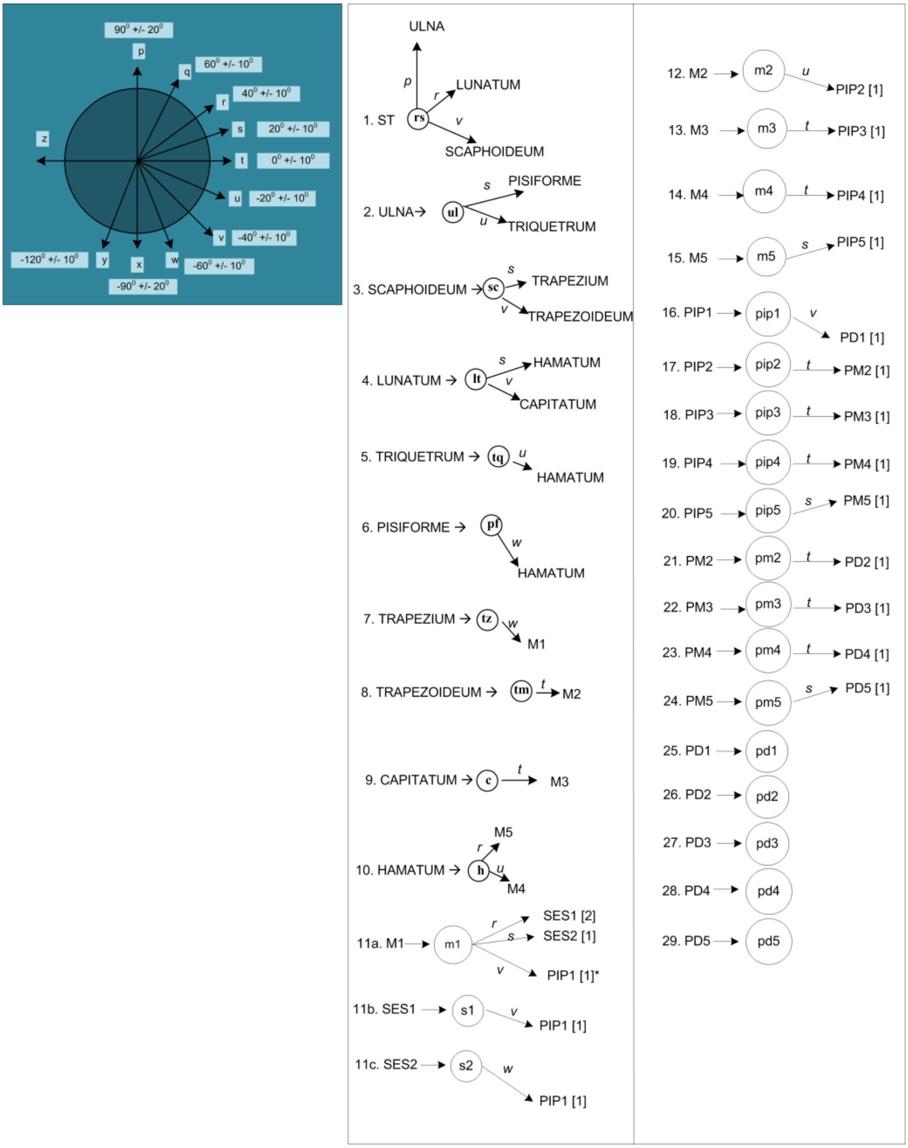


Fig. 2. The Γ_{hand} graph defining relations between hand bones and a set of productions P_{hand}

The proposed linguistic formalism for semantically analysing image data has also been used to propose a formal description of the analysis of biometric hand features. These features, with the added elements of the semantic data analysis, make an extended identification analysis possible. A formal solution based on defining a set of biometric characteristic features of the hand is proposed for analysing the biometric features of bones:

$$B_{hand} = \{th_{ij}, l_{ij}, s_{ij\text{-}ij}, th_{mi}, l_{mi}, s_n\}$$

where:

th_{ij} – denotes the thickness of the bones of the i^{th} finger and the j^{th} phalanx i = {I, II, III, IV, V}, $j = \{1, 2, 3\}$
l_{ij} – denotes the length of the bones of the i^{th} finger and the j^{th} phalanx
$s_{ij\text{-}ij}$ – is the size of areas between individual hand bones
th_{mi} – is the thickness of the i^{th} metacarpus bone
l_{mi} – is the length of the i^{th} metacarpus bone
s_n – is the size of wrist bones

The elements of the B_{hand} set are presented in Figure 3.

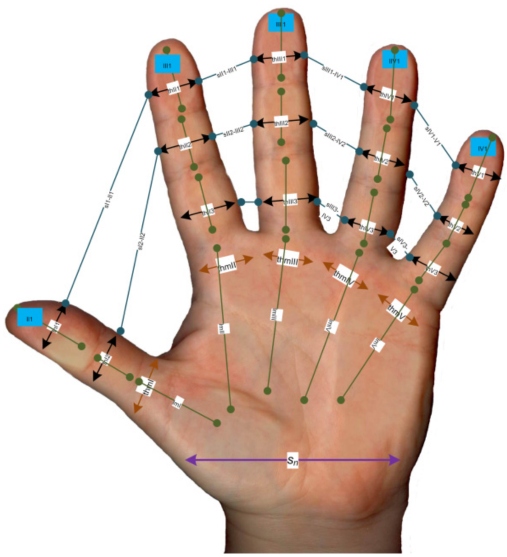


Fig. 3. Elements of the set B_{hand}

The following elements describing the shapes of handprints have been added to the proposed set B_{hand} defining the biometric features of the hand:

$$L_{hand} = \{p_i, o_j\}, i = \{1,...,5\}, j = \{1,...,3\}$$

where:

p_i – denotes the print of the i^{th} finger of the hand (from one to five)
o_j – the shape of one of the three biometric prints of the palm

The elements of the L_{hand} set are presented in Figure 4.

Combining the sets B_{hand} and L_{hand} yields the set BL_{hand} (Fig. 5), having the following form:

$$BL_{hand} = \{th_{ij}, l_{ij}, s_{ij\text{-}ij}, th_{mi}, l_{mi}, s_n, p_i, o_j\}$$

where the individual elements of this set are defined as above.

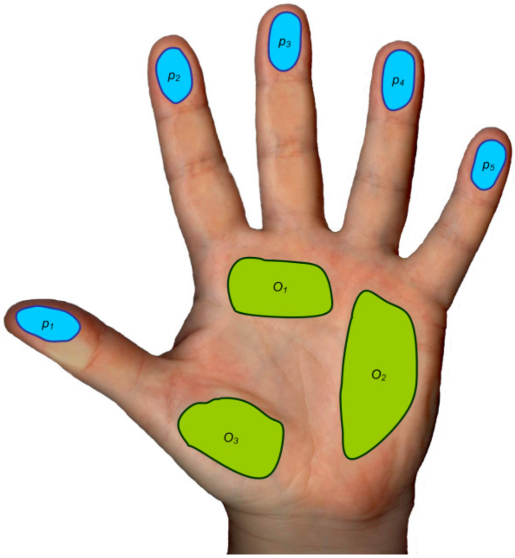


Fig. 4. Elements of the set L_{hand}

The BL_{hand} set defines the biometric features and shapes of handprints as well as the shape of fingerprints, which make it possible to conduct a biometric analysis. Data on biometric features stored in the E-UBIAS system supports personal identification and personal verification.

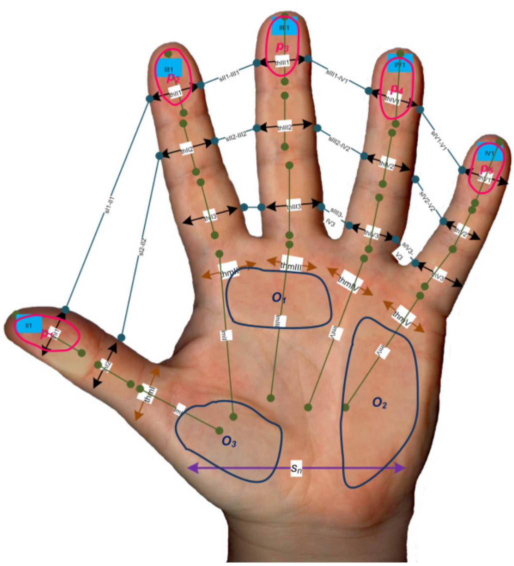


Fig. 5. Elements of the set BL_{hand}

3 DNA Cryptography in E-UBIAS Systems

E-UBIAS systems enhanced with DNA cryptography and used to generate keys based on DNA codes belonging to each person produce a vector of the biometric features of a given person. Every person has his/her unique DNA code which can describe him/her.

The genetic code has the form shown in Fig. 6.

	U (Uracyl)	**C (Cytosine)**	**A (Adenine)**	**G (Guanine)**
U (Uracyl)	UUU Phenylalanine *Phe* UUC Phenylalanine *Phe* UUA Leucine *Leu* UUG Leucine *Leu*	UCU Serine *Ser* UCC Serine *Ser* UCA Serine *Ser* UCG Serine *Ser*	UAU Tyrosine *Tyr* UAC Tyrosine *Tyr* UAA *Stop* UAG *Stop*	UGU Cysteine *Cys* UGC Cysteine *Cys* UGA *Stop* UGG Tryptophan *Trp*
C (Cytosine)	CUU Leucine *Leu* CUC Leucine *Leu* CUA Leucine *Leu* CUG Leucine *Leu*	CCU Proline *Pro* CCC Proline *Pro* CCA Proline *Pro* CCG Proline *Pro*	CAU Histidine *His* CAC Histidine *His* CAA Glutamine *Gln* CAG Glutamne *Gln*	CGU Arginine *Arg* CGC Arginine *Arg* CGA Arginine *Arg* CGG Arginine *Arg*
A (Adenine)	AUU Isoleucine *Ile* AUC Isoleucine *Ile* AUA Isoleucine *Ile* AUG Methionine *Met*	ACU Threonine *Thr* ACC Threonine *Thr* ACA Threonine *Thr* ACG Threonine *Thr*	AAU Asparigine *Asn* AAC Asparigine *Asn* AAA Lysine *Lys* AAG Lysine *Lys*	AGU Serine *Ser* AGC Serine *Ser* AGA Arginine *Arg* AGG Arginine *Arg*
G (Guanine)	GUU Valine *Val* GUC Valine *Val* GUA Valne *Val* GUG Valine *Val*	GCU Alanine *Ala* GCC Alanine *Ala* GCA Alanine *Ala* GCG Alanine *Ala*	GAU Aspartate *Asp* GAC Aspartate *Asp* GAA Glutamate *Glu* GAG Glutamate *Glu*	GGU Glycine *Gly* GGC Glycine *Gly* GGA Glycine *Gly* GGG Glycine *Gly*

Fig. 6. Genetic code

In addition, an individual medium containing the DNA genetic information can be combined with biometric features described by the BL_{hand} set. The nucleotide polymer (DNA) is made up of purine bases, namely adenine (A) and guanine (G), pyrimidine bases - cytosine (C) and thymine (T), deoxyribose nucleotides and phosphate groups. The bases are bonded in pairs according to the following diagram (Fig. 7):

A-T (A-U), G-C, T-A (U-A), C-G

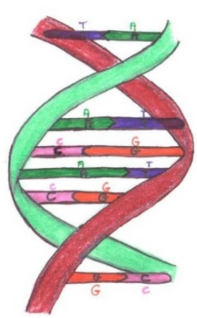


Fig. 7. DNA chain (code)

The bonding between purine and pyrimidine basses allows the DNA code to be written in an unanimous form, and the DNA code itself can be used for biometric analyses. The encoded DNA information forms the basis of cryptographic, identification and verification analyses.

This bonding has been adopted for defining a set which can for be used for executing the biometric, identification and verification analysis:

$$GEN\text{-}BL_{hand} = \{\text{DNA CODE (A-T G-C T-A C-G)}, th_{ij}, l_{ij}, s_{ij\text{-}ij}, th_{mi}, l_{mi}, s_n, p_i, o_j\}$$

Genetic information coding is aimed at making its contents secret. The $GEN\text{-}BL_{hand}$ set with the coded form of genetic information takes the following form:

$$GEN(CODE)\text{-}BL_{hand} = \{010110011000101000, th_{ij}, l_{ij}, s_{ij\text{-}ij}, th_{mi}, l_{mi}, s_n, p_i, o_j\}$$

As a result of DNA information coding, the personal identification and the cognitive analysis carried out by E-UBIAS systems, the feature vectors assigned to a given person in biometric data analysis systems can contain information about (Fig. 8):

- the DNA code;
- biometric features;
- physical features;
- hand bone deformations and pathologies.

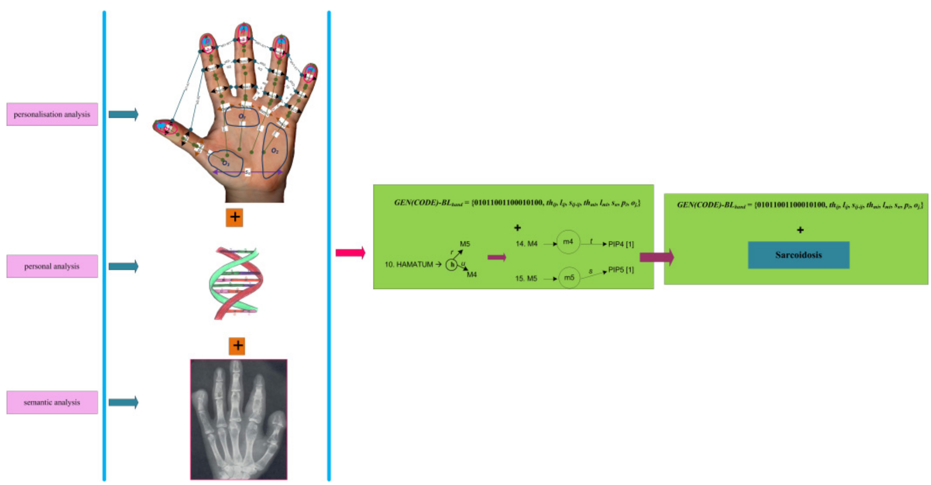


Fig. 8. A biometric data analysis system supplemented with semantic analysis executed in E-UBIAS systems

Figure 8 presents a description of the operation of biometric data analysis systems supplemented with elements of the semantic analysis of image data. This type of analysis makes it possible to assign a given person his/her genetic information in a coded form, biometric data, physical features and possible lesions. This multi-stage process of personal identification significantly reduces the possibility of an error due to the incorrect process of reasoning and personal verification.

The above vectors may also contain information on other lesions of a given person, e.g. those in coronary arteries, deformations of long bones of extremities and lesions in internal organs. Such information can be used for an in-depth data analysis which is not carried out by traditional verification systems, but which can, in individual cases, be used for complex personal identification problems.

4 Conclusion

The idea of biometric data analysis in personal identification and verification systems carried out using E-UBIAS cognitive systems presented here allows analysing and interpreting complex (very extensive) data sets. At the same time, systems of this kind can execute not just a simple identification analysis based on biometric features stored in the system (like the shape of finger/handprints or an analysis of the iris), but can analyse various data sets. This variety of data also allows various analysis methods to be employed. The ability to collect information on biometric features, physical features, structural features of individual human organs, the occurrence of lesions as well as the DNA code in the knowledge bases of cognitive information systems (in a form coded and thus inaccessible to the typical system user) offers opportunities of enhancing E-UBIAS systems with freely chosen data and information.

Processes of modelling data stored in E-UBIAS systems allow the personal data kept in E-UBIAS cognitive systems to be made completely secret.

Acknowledgement. This work has been supported by the National Science Centre, Republic of Poland, under project number N N516 478940.

References

1. Cohen, H., Lefebvre, C. (eds.): Handbook of Categorization in Cognitive Science. Elsevier, The Netherlands (2005)
2. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. A Wiley-Interscience Publication John Wiley & Sons, Inc. (2001)
3. Meystel, A.M., Albus, J.S.: Intelligent Systems – Architecture, Design, and Control. A Wiley-Interscience Publication John Wiley & Sons, Inc., Canada (2002)
4. Ogiela, L.: Cognitive Systems for Medical Pattern Understanding and Diagnosis. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part I. LNCS (LNAI), vol. 5177, pp. 394–400. Springer, Heidelberg (2008)
5. Ogiela, L.: Syntactic Approach to Cognitive Interpretation of Medical Patterns. In: Xiong, C.-H., Liu, H., Huang, Y., Xiong, Y.L. (eds.) ICIRA 2008, Part I. LNCS (LNAI), vol. 5314, pp. 456–462. Springer, Heidelberg (2008)
6. Ogiela, L.: UBIAS Systems for Cognitive Interpretation and Analysis of Medical Images. Opto-Electronics Review 17(2), 166–179 (2009)
7. Ogiela, L.: Computational Intelligence in Cognitive Healthcare Information Systems. In: Bichindaritz, I., Vaidya, S., Jain, A., Jain, L.C. (eds.) Computational Intelligence in Healthcare 4. SCI, vol. 309, pp. 347–369. Springer, Heidelberg (2010)

8. Ogiela, L.: Cognitive Informatics in Automatic Pattern Understanding and Cognitive Information Systems. In: Wang, Y., Zhang, D., Kinsner, W. (eds.) Advances in Cognitive Informatics and Cognitive Computing. SCI, vol. 323, pp. 209–226. Springer, Heidelberg (2010)
9. Ogiela, L., Ogiela, M.R. (eds.): Cognitive Techniques in Visual Data Interpretation. SCI, vol. 228, pp. 29–39. Springer, Heidelberg (2009)
10. Ogiela, L., Ogiela, M.R.: Semantic Analysis Processes in UBIAS Systems for Cognitive Data Analysis. In: You, I., Barolli, L., Tang, F., Xhafa, F. (eds.) IMIS 2011 - 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, June 30-July 2, pp. 182–185. Korean Bible University, Seoul (2011)
11. Ogiela, L., Ogiela, M.R.: Semantic Analysis Processes in Advanced Pattern Understanding Systems. In: Kim, T.-H., Adeli, H., Robles, R.J., Balitanas, M. (eds.) AST 2011. CCIS, vol. 195, pp. 26–30. Springer, Heidelberg (2011)
12. Ogiela, L., Ogiela, M.R.: Advances in Cognitive Information Systems. COSMOS, vol. 17. Springer, Heidelberg (2012)
13. Zhong, N., Raś, Z.W., Tsumoto, S., Suzuki, E. (eds.): 14th International Symposium on Foundations of Intelligent Systems. ISMIS, Maebashi City (2003)

Path-Oriented Integration Method for Complex Trees

Marcin Maleszka and Ngoc Thanh Nguyen

Institute of Informatics, Wroclaw University of Technology,
Wybrzeze Wyspianskiego 27, 50-370 Wroclaw
{marcin.maleszka,ngoc-thanh.nguyen}@pwr.wroc.pl

Abstract. The hierarchical data format becomes increasingly common. One of the problems arising with it is efficient tree integration, i.e. integration of XML databases. One possible approach is path-based integration, used among others in applications using XQuery and XPath database queries. This article offers a generalization for integration tasks based on this approach, as well as a short analysis of the result in terms of other existing integration criteria. The paper presents estimations for such criteria as Completeness, Minimality and Precision.

Keywords: tree integration, integration criteria, algorithms, query integration.

1 Introduction

With the increasing application of hierarchical data formats in modern world, the research focused on this kind of structures becomes increasingly important. The XML data format and the various hierarchical ontologies are the most common examples. In such systems, a variety of ways to integrate trees exists, based on various aims or criteria [9].

In our previous work we have explored some of the general properties for such criteria as Completeness or Minimality [8]. One of the results was the possibility of multi-stage integration in Completeness-based problems (by "criterion-based integration" we mean an integration aimed at that criterion retaining its maximum level), which allows both the reduction of computational complexity for some tasks and a distributed approach to the problem.

In this paper we explore a specific type of Completeness-based tree integration tasks, that is path-oriented integration (integration guarantees that the criterion Path Completeness is equal to 1, for definition of Path Completeness see Section 4). This finds common use in such practical application as distributed XML databases, where the query is described by the XQuery or XPath constructs. The aim for the integration is for the system to have the same query answers after the process, as it did before – in other terms, for all the root-leaf paths in the tree to be preserved. Various approaches to this task exist and here we present a generalization of them. With the mathematical apparatus we proposed in [10] we can use this generalized algorithm to determine the properties of the algorithm and the desired solution.

This generalized algorithm, thanks to the analysis conducted in our research, finds immediate use in several multi-agent applications. The most important result in this area is the fact that distributed integration will give the same result as the centralized

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 84–93, 2012.

approach. This can be used in many existing agent systems, allowing for a certain degree of decentralization. We also started a follow-up research project in the area of distributed, agent-based knowledge bases.

This paper is organized as follows: Section 2 contains a short overview of related research in the general area of tree integration and of some existing approaches to path-based integration; Section 3 introduces the mathematical terminology and structures used in this paper; Section 4 contains the generalized integration algorithm and its analysis in terms of criteria; Section 5 concludes the paper with remarks on possible future work.

2 Related Works

The first research on tree integration was conducted in the 1980s in [1] and other works of evolutionary biologists. In those works the aim was to resolve the conflicting results of different experimental methods and create a single evolutionary tree out of multiple inputs of that type. The structure researched was mathematically the n-tree, that is a tree with only the leafs labeled, and the single criterion used was what was later called Optimality [11][12].

Other work into tree integration was done under the general area of schema matching and ontology mapping. In the former case [13] proposes three general criteria for integrating XML schemas, following earlier works in the area: *completeness* (all elements from the input schemas remain in the output schema), *minimality* (each unique element is defined only once in the output schema) and *understandability* (multiple interpretation, most commonly: a proper formatting of the output). While those three criteria are based on the work done for schema integration, authors modify the definitions to work with a specifically created hierarchical structure. In the case of ontology integration, the authors of [15] propose the following: *legibility* (with two sub-criteria: *minimality* – every element appears only once; *clarity* – it is easy to read by the user), *simplicity* (a minimal possible number of elements occur), *correctness* (all elements are properly defined), *completeness* (all relevant features are represented) and *understandability* (similar to *clarity*, but focuses on navigating the structure). Additionally, [5] adds *importance* as a measure more related to the concepts in the ontology, but which may be also used to compare the results of the integration with the inputs (as in the methodology used here).

The path-oriented integration of trees was researched since the introduction of such XML derived applications as XQuery and XPath database search queries [14][3]. The most common case of path-oriented integration is required in distributed XML systems (i.e. data warehouses) operating based on the queries of that type. In that case the integration is only virtual (a common schema is adopted on the input databases) with the aim of allowing access to all the possible query paths from all the inputs. Examples of this may be found in [2][3][7] and more. The integration itself may be conducted in a variety of ways, including a top-down approach of tree inclusion of [3] aimed at lowering memory requirements, the twig stack approach of [2] aimed at lowering CPU load, navigation driven approach of [14] using a specific algebra for the implemented platform and other algebra-based solutions [4][6].

3 Complex Tree Integration

In our research, here and in earlier papers [8][9][10] we use a specific structure called the complex tree, that is a labeled tree with more complex vertices and some additional schema information. The structure was developed in order to model most standard hierarchical structures, like XML (and its schemas), n-trees, and some basic ontologies. Each of these structures may be transformed into a complex tree and then all the properties presented here occur. After the integration a complex tree may be transformed back into the former structure, without any loss. In this section we present the definition of the complex tree structure, as well as the definition of the Integration Task, defined as in [8].

3.1 Complex Tree

The complex tree was first defined in [10] with further refinements done in [9]. Additional research done in [8] shown that the use of multiset definition allows for more precise results. The definition presented below is the one shown in [8]:

Definition 1: complex tree (CT) is a tuple CT = (T,S,V,E), where:

- T – a multiset of node types T = {t_1, t_2, t_3, … };
- S – a function determining a sequence of attributes for each type of node $S(t_i) = (a_{i1}, a_{i2}, ..., a_{in_i})$;
- V – a multiset of nodes (vertices), where each node $v \in V$ is a triple v=(l, t, A) in which the l is the label of the node, $t \in T$ is the type of the node and A is the sequence of attributes and their values for this node (as determined by the S function);
- E – a multiset of edges, $E = \{e = (v, w) : v \in V, w \in V\}$.

This definition of the complex tree is very general. Practical uses differentiate between complex tree schema and complex tree instance (i.e. as generalizations of XSD and XML). While the rest of this paper is based on this general definition, please note that these specific interpretations – shortly described below – are used in standard tasks and algorithms.

Complex tree schema (CTS), like database schema or XML Schema, is a structure that describes the possible outlook of multiple structure it is representing - complex tree instances (CTI). As such, the CTS has less actual nodes and no attribute values – no actual data. On the other hand the CTS may have some types that will not occur in the instance of this schema [8][9].

Complex tree instance, like the actual database table or XML document, is a structure that stores the data or knowledge. As such, it must not have all the types from the schema, but all the attributes must have determined values (even if it represents unknown) and the structure of the nodes may become very complex [8][9].

We represent the i-th complex tree schema in an ordered set as $CT^{(i)}$, where:

$$CT^{(i)} = (T^{(i)}, S^{(i)}, V^{(i)}, E^{(i)}) .$$

Each CTS may have multiple complex tree instances, we represent the j-th CTI of i-th CTS as $CT^{(i)}{}_{(j)}$, where:

$$CT^{(i)}{}_{(j)} = (T^{(i)}{}_{(j)}, S^{(i)}{}_{(j)}, V^{(i)}{}_{(j)}, E^{(i)}{}_{(j)}) .$$

For ease of readability we will also use the following notation to indicate CTIs:

$$CT_j = (T_j, S_j, V_j, E_j) .$$

This notation will be used in the rest of this paper. Note, that the same applies to CTS.

3.2 Integration Task for CTI

As the integration of both CTI and CTS are defined in an identical manner, here we present only the Integration Task for CTI. Details on the differences between both Tasks and the definition of CTS integration may be found in [9][10]. The Integration Task for CTI is defined as follows [8]:

The input of the integration process is n complex trees $CT_1, CT_2, \ldots, CT_n$ from the same schema.

$$CT_1 = (T_1, S_1, V_1, E_1), \ldots, CT_n = (T_n, S_n, V_n, E_n) .$$

The output of the integration process is one complex tree CT^*, connected with input structures by a group of criteria.

$$CT^* = (T^*, S^*, V^*, E^*) .$$

The parameters of the integration task are the integration criteria $K=\{K_1,K_2,\ldots K_m\}$ tying CT^* with $CT_1, CT_2,\ldots,CT_n$, each at least at a given level $\alpha_1,\ldots,\alpha_n$

$$K_i: M_j(CT^* | CT_1, CT_2, \ldots, CT_n) \geq \alpha_i ,$$

where M_j is a measure of a criterion (as defined in [9]). The specific notation of the measure arguments was selected for ease of readability and is used thorough this paper.

4 Path-Oriented Integration of Complex Trees

For purposes of determining the properties of XQuery/XPath based integration, we have adopted similar structures and terminology in our research. In [9] we introduced the criterion of Path Completeness that allows the correct aim of the integration and in our current research was used to determine those properties. The Path Completeness definition and the supporting definition of a path in a complex tree are presented below, as defined in [9]:

Definition 2: A path in a complex tree is a sequence of nodes, starting with the root of the tree, and ending with a leaf of the tree, where each next element in the sequence is a child of the previous element.

$$p = \{v_{root,v_1,\ldots,v_{n_p}}\}$$

The number of paths in a complex tree is equal to the number of leaves. Let $P_{CT}(CT)$ be a set of all paths in CT. Let p_i be the *i*-th element in the sequence *p*.

Definition 3: Path-based Completeness is a measure representing how many of the paths, or parts of those paths, from the input trees occur in the integrated tree.

$$C_{path}(CT^*|CT_1, CT_2) = \frac{1}{card\{P_{CT}(CT_1) \cup P_{CT}(CT_2)\}} \sum_{p \in P_{CT}(CT_1) \cup P_{CT}(CT_2)} l_p(p.P_{CT}(CT^*))$$

where:

- $l_p(p,P_{CT}(CT))$ measures the length of the path (or part of the path) in $P_{CT}(CT)$ most similar to *p*,

- $l_p(p, P_{CT}(CT)) = \frac{1}{n_p+1} max_{q \in P_{CT}(CT)} \sum_{i=0}^{n} s(p_i, q_i)$

- $s(p_i, q_i) = \begin{cases} 1, if\ p_i = q_i \\ 0, otherwise \end{cases}$

As we detailed in [9] Path-based Completeness is equal to 1 if all the paths from CT_1 and CT_2 occur in CT*. It is also equal to 0, if none of those paths occur in CT*. Note that this allows also for partial paths. Here we will use a more strict definition, with $l_p = n_p$ if path occurs as a whole and $l_p = 0$ otherwise.

4.1 Generalized Algorithm for Path-Oriented Integration

Based on the algorithms proposed in the literature (Section 2) we propose a generalized algorithm for complex trees. This is the most basic version of path-oriented integration and while it is not optimized for use in applications, the simple structure allows a mathematical analysis of the result.

```
Input: N complex trees
Output: one integrated complex tree CT

BEGIN
1. Create set P of all paths in input trees
2. Create the output complex tree CT by using random
   path p from P as its template. Remove p from P.
3. While card(P)>0
   3.1. Get random path r from P
   3.2. If r starts with the same node as the root of CT
      3.2.1.Compare the subsequent nodes in r with
            subsequent children of CT’s root, until a
            node not occurring in CT occurs in r or
            until the last element of r
      3.2.2.Attach the rest of r as a subtree of the
            last identical node in CT
   3.3. Otherwise
      3.3.1.If a node identical to the first node in r
            also occurs in CT
         3.3.1.1.Attach the rest of r at that location
      3.3.2.Otherwise
         3.3.2.1.Attach r as a subtree of CT’s root
   3.4. Remove r from P
END
```

This basic algorithm has the computational complexity is in the worst case $O(P^2)$, where P is the overall number of paths in all input trees. This may also be expressed as $O(N^2)$, where N is the number of all nodes in the input trees. Some algorithms in the literature are faster (but never linear), but analyzing the other integration criteria for them is impossible due to the methods applied.

4.2 Algorithm Properties

A series of criteria other than Path Completeness described above was considered to help analyze the properties of the proposed algorithm. Those criteria were earlier described in [9] and will be shortly presented here as a basis for the analysis. Note that the algorithms discussed in Section 2 fulfill these properties and are optimized for specific applications (i.e. lower memory requirements, lower CPU requirements).

One of the considered groups of criteria were the other types of completeness, as detailed in [9]. These consist of:

Definition 4: Structure Completeness is a measure representing the number of nodes (identified by types and labels) from the input structures which remain after the integration.

$$C_D(CT^*|CT_1, CT_2) = \frac{1}{card(V_1 \cup V_2)} \sum_{v \in V_1 \cup V_2} m_V\,(v, V^*)$$

where:

$$m_V(v, V) = \begin{cases} 1, if\ v \in V \\ 0, if\ v \notin V \end{cases}$$

Definition 5: Relationship Completeness is a measure representing how many of the relationships from the input trees occur in the integrated tree.

$$C_{rel}(CT^*|CT_1, CT_2) = \frac{1}{card(E_1 \cup E_2)} \sum_{e \in E_1 \cup E_2} m_E\,(e, E^*)$$

where

$$m_E(e, E) = \begin{cases} 1, if\ e \in E \\ 0, if\ e \notin E \end{cases}$$

The other criteria considered in this analysis are:

Definition 6: Minimality is a basic measure which enables comparing the size of integrated tree with input trees, where the output tree should be no larger than the sum of the input trees.

$$M_V(CT^*|CT_1, CT_2) = \begin{cases} 1, if\ card(V^*) \leq \max\{card(V_1), card(V_2)\} \\ \frac{\max\{card(V_1), card(V_2)\}}{card(V^*)}, otherwise \end{cases}$$

Definition 7: Precision is a measure representing how many new elements were introduced during the integration and if no duplicate information is in the output tree.

Here, only vertices are considered.

$$P_V(CT^*|CT_1, CT_2) = \frac{card(V_1 \cup V_2)}{card(V^*)}$$

With the criteria defined, a short analysis of the algorithm in these terms will be presented.

First we will consider the set interpretation of both Structure and Relationship Completeness. As described in [9], the Path Completeness equal to 1 makes it necessary for both the Relationship Completeness and the Structure Completeness to be equal to 1. While this is mathematically strict, we have also run a number of simulations with the set interpretation of the Criteria to observe the phenomenon.

As further research has shown that multiset interpretation of criteria is closer to real world applications, we have repeated the simulations for this modification. The results of the tests (averaged over a million runs) have shown that the other criteria are not necessarily equal to 1, when the algorithm designed for Path Completeness equal to 1 is used. For example, for two input trees of size10 (number of nodes), with identical roots and with the labels chosen among the set {1,2,…,100}, the

Table 1. Excerpt of the simulation run results. T = no. of input trees. N = size of input tree. M = range of labels. I = identical root of input trees (yes, no). S = multiset criteria (yes/no).

	T=2, N=10, M=100, I=y, S=y	**T=2, N=10, M=100, I=n, S=y**	**T=2, N=10, M=100, I=y, S=n**	**T=2, N=10, M=20, I=y, S=y**	**T=20, N=10, M=100, I=y, S=y**
Structure Completeness	0,92	0,99	1	0,91	0,85
Relationship Completeness	0,99	0,99	1	0,98	0,98
Path Completeness	1	0,49	1	1	1
Minimality	0,69	0,58	0,69	0,69	0,87
Precision	1	0,91	1	1	1

Relationship Completeness was equal to 0,99 and the Structure Completeness was equal to 0,92 (for more results please refer to Table 1). This result is close to the results of the following assessment:

Let us assume that M is the set of all possible labels, while E is the set of labels actually used in two different input trees (with the same root). N_1 and N_2 are the number of nodes in both trees, respectively, and L_1 and L_2 are the number of leafs in them. From the tree definition we have that:

$$L_1 \leq N_1 - 1; L_2 \leq N_2 - 1 \tag{1}$$

$$\max\{N_1, N_2\} \leq |E| \leq N_1 + N_2 \tag{2}$$

We can also estimate that the probability of a label appearing in a distinct tree at a given position is 1/E (in a more general situation it is 1/M, but here this estimation is not necessary). Thus, considering that the integration algorithm may join the nodes to reduce their number, but will not do so to leafs, we can estimate Structure Completeness as:

$$C_S = \frac{(N_1 - L_1)(N_2 - L_2)}{E^2} + \frac{L_1 + L_2}{E}$$

Which may be further extended to:

$$C_S = \frac{N_1 N_2 + L_1 L_2 + (E - N_1)L_2 + (E - N_2)L_1}{E^2}$$

Using the assumption made in (1) and (2), we can see that this has an upper bound:

$$C_S \leq \frac{N_1 N_2 + (N_1 - 1)(N_2 - 1) + (E - N_1)(N_2 - 1) + (E - N_2)(N_1 - 1)}{E^2}$$

Which in turn can be simplified to:

$$C_S \le \frac{N_1 + N_2 - 2}{E} + \frac{1}{E^2}$$

Let us now use this upper bound for the least optimal case of integrating two trees with the proposed algorithm, with the same parameters as the experimental run described above. This means that $N_1=N_2=10$ and $E=19$ (the root is identical). The upper bound in this case is:

$$C_S \le \frac{10 + 10 - 2}{19} + \frac{1}{19^2} \cong 0{,}95$$

Note that this will occur in multiple cases of integration, but not in all of the runs. If more nodes are identical in the trees the result is $C_S \le 1$.

While this does not provide an exact result, this upper bound approximation allows the estimation of possible data loss or compression (depending on the application). Furthermore, the same estimation may be conducted for other parameters and for Relationship Completeness.

The same estimation may be conducted for other criteria. Due to space constraints in this paper, we only provide an overview of other properties below:

- for multiple input trees with different root, Path Completeness is no larger than one over the number of input trees (where common sense would dictate that this criterion is equal to that value),
- precision is always maximized if the roots are identical, for different roots the precision is lower by the number of identical nodes in input trees over the number of nodes in the output tree,
- using set-based criteria for calculations yields higher values of these criteria (the practical use of such criteria is negligent),
- the output tree is no larger (in terms of number of nodes) than the sum of input trees and usually only up to 50% larger than the largest input tree.

5 Future Works

The research presented in this paper is the element of a larger project to determine the properties of all cases of tree integration, basing on the adopted structure of complex tree. Our earlier findings were presented in [10] and consisted of some general properties for Completeness, Minimality and Precision. Consecutively, in this paper we present some of the properties of Path Completeness based integration of complex trees. Our future research will be aimed to determine more properties of tree integration, including presenting generalized algorithms presenting these properties.

Acknowledgments. This work was partially supported by the Polish Ministry of Science and Higher Education under the grant no. N N519 444 939 (2010-2013).

References

1. Barthelemy, J.P., McMorris, F.R.: The Median Procedure for n-Trees. Journal of Classification 3, 329–334 (1986)
2. Bruno, N., Koudas, N., Srivastava, D.: Holistic Twig Joins: Optimal XML Pattern Matching. In: ACM SIGMOD, Madison, Wisconsin, USA, June 4-6 (2002)
3. Chen, Y., Shi, Y., Chen, Y.: Tree Inclusion Algorithm, Signatures and Evaluation of Path-Oriented Queries. In: SAC 2006, Dijon, France, April 23-27 (2006)
4. Delobel, C., Reynaud, C., Rousset, M.-C., Sirot, J.-P., Vodislav, D.: Semantic integration in Xyleme: a uniform tree-based approach. Data & Knowledge Engineering 44, 267–298 (2003)
5. Duong, T.H., Jo, G.S., Jung, J.J., Nguyen, N.T.: Complexity Analysis of Ontology Integration Methodologies: A Comparative Study. Journal of Universal Computer Science 15(4), 877–897 (2009)
6. Fan, W., Garafalakis, M., Xiong, M., Jia, X.: Composable XML Integration Grammars. In: CIKM 2004, Washington, DC, USA, November 8-13 (2004)
7. May, W., Behrends, E., Fritzen, O.: Integrating and querying distributed XML data via XLink. Information Systems 33, 508–566 (2008)
8. Maleszka, M., Nguyen, N.T.: Some properties of complex tree integration criteria. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part II. LNCS, vol. 6923, pp. 1–9. Springer, Heidelberg (2011)
9. Maleszka, M., Nguyen, N.T.: A Method for Complex Hierarchical Data Integration. Cybernetics and Systems 42(5), 358–378 (2011)
10. Maleszka, M., Nguyen, N.T.: A Model for Complex Tree Integration Tasks. In: Nguyen, N.T., Kim, C.-G., Janiak, A. (eds.) ACIIDS 2011, Part I. LNCS, vol. 6591, pp. 36–46. Springer, Heidelberg (2011)
11. Nguyen, N.T.: Advanced Methods for Inconsistent Knowledge Management. Springer, London (2008)
12. Nguyen, N.T.: Inconsistency of Knowledge and Collective Intelligence. Cybernetics and Systems 39(6), 542–562 (2008)
13. Passi, K., Lane, L., Madria, S., Sakamuri, B.C., Mohania, M., Bhowmick, S.: A Model for XML Schema Integration. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2002. LNCS, vol. 2455, pp. 193–202. Springer, Heidelberg (2002)
14. Papakonstantinou, Y., Vassalos, V.: Architecture and Implementation of an XQuery-based Information Integration Platform. Bulletin of the Technical Committee on Data Engineering 25(1), 18–26 (2002)
15. Trinkunas, J., Vasilecas, O.: Ontology Transformation: from Requirements to Conceptual Model. Scientific Papers, University of Latvia. Computer Science and Information Technologies, vol. 751, pp. 52–64 (2009)

Integrating Quantitative Attributes in Hierarchical Clustering of Transactional Data

Mihaela Vranić, Damir Pintar, and Zoran Skočir

University of Zagreb,
Faculty of Electrical Engineering and Computing
{mihaela.vranic,damir.pintar,zoran.skocir}@fer.hr

Abstract. Appropriate data mining exploration methods can reveal valuable but hidden information in today's large quantities of transactional data. While association rules generation is commonly used for transactional data analysis, clustering is rather rarely used for analysis of this type of data. In this paper we provide adaptations of parameters related to association rules generation so they can be used to represent distance. Furthermore, we integrate goal-oriented quantitative attributes in distance measure formulation to increase the quality of gained results and streamline the decision making process. As a proof of concept, newly developed measures are tested and results are discussed both on a referent dataset as well as a large real-life retail dataset.

Keywords: Transactional Data, Hierarchical Clustering, Quantitative Attributes, Distance Measures, Retail Data.

1 Introduction

There are large quantities of transactional data being stored in today's business and science information systems. After eventual extraction of summarized data, transactional data remains largely unused in a decision making process, even though it may potentially contain useful information which can only be discovered by analyzing transactions in detail instead of just using the aggregated form. Association rules generation is a well known data mining technique for detailed examination of data in unsummarized, raw transactional form. However, due to some of its drawbacks concerning its resource requirements and result presentation it is not used as often as it could be. Additionally, quantitative attributes of transactional data and proper management of them presents a special issue in itself with plenty of current research dealing with this particular problem.

In this paper we present how basic parameters connected to association rules generation can be adopted to be used in hierarchical clustering of transactional elements. We further include quantitative attributes in distance measure formulas. The idea behind this is the fact that the analyst often has a definite quantifiable goal in mind and is trying to come up with decisions which will result in raising the profit, getting more sales or achieving better performance results. By integrating quantitative attributes which directly influence the goal

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 94–103, 2012.

of the research in the data mining algorithm, we can hope to gain results which can be more easily interpreted in the context of achieving the desired goal.

When it comes to transactional data analysis we focused on well known and most widely used parameters usually associated with association rules generation - support and confidence. We transformed and adapted them to represent distance more accurately and then integrated quantitative attributes. Additionally, we have also included asymmetric binary dissimilarity (presented in [1]), a measure developed to express dissimilarity between two asymmetric binary variables.

This paper is organized as follows: Section 2 presents related work. Existing methods and research are briefly presented. Section 3 discusses the method of hierarchical clustering and adaptation of distance measures to reflect important parameters of items' relations and incorporate quantitative attributes. Section 4 shows the application of constructed measures on two real-life input datasets - one being rather small and thus convenient for examination of differences between gained results with different chosen measures and the other being real life large dataset showing possible usage in real life retail business. Finally, Section 5 offers final insights and conclusions on the issues presented in this paper together with future work.

2 Related Work

One of the most influential works in the field of descriptive data mining analysis focused on transactional data is [2]. It gives foundations for association rules generation, a widely accepted method for transactional data research. During the years of its usage, certain deficiencies were noticed which remained only partially solved. Main are: resource demanding algorithms, large number of uninteresting rules ([3], research concentrated on finding only interesting ones are e.g. [4], [5], [6]), presentation of rules stemming from the same relationship in different rules spread through whole model. Some recent papers argue that very often the existence of frequent itemset (collections of transactional elements satisfying certain parameters) in itself presents an interesting revelation which can be acted upon. Additionally, their presentation eliminates the potential problem of implying causality [7].

Hierarchical clustering with usage of appropriate distance measures would enable clear presentation of relationships between transactional elements. This method relies on calculation of distances between pairs of elements and thus is not very resource demanding. It enables formation of concise and easy interpretable structures (dendrograms) which reveal object groupings in hierarchical manner and are easily interpretable by business analysts. For appropriate interpretation of closeness of analyzed objects, or in our case transactional elements, distance measure formulation is the key issue. Our work presented in [8] gives good insight how to model a distance measure between observations which are described by a mixture of variable types. In [9], [10] further research is given with transactional data as a main focus of interest.This paper continues this research as we augment parameters used in native method for transactional data exploration with additional quantitative attributes.

Association rules generation also deals with the problem of quantitative attributes integration (like [11], [12]). Due to the completely different approach, our integration of quantitative attributes that uphold the main goal of the analyst successfully bypasses some problems addressed in association rules discovery. Here we directly use those quantitative attributes without the need to divide them in separate classes. However, we used some findings presented in [13] to elevate the modeling of dendrograms which include items' categories for large real life retail dataset presented in Section 4.

3 Hierarchical Clustering Adaptation

3.1 Adapting Measure for Hierarchical Clustering

Hierarchical clustering algorithms rely on calculating distance (as opposed to closeness) between pairs of objects. Distance calculation is usually based on specializations of Minkowski distance, most common being the Euclidian or Manhattan distance measure. This works well for variables using interval or relational scale, but is not particularly suited to transactional data, which is basically a collection of binary vectors denoting presence or absence of observed items in each noted transaction.

Our initial research dealt with procuring distance measures oriented specifically toward transactional data, resulting in hierarchical clustering becoming a feasible alternative (or complement) to association rule generation. The idea was to find measures commonly associated with transactional data analysis and adapt them to represent distance. Our final choices were measures of support, confidence and asymmetric binary dissimilarity.

Before delving into specifics of each measure and its subsequent adaptation, it is important to define expected properties of a distance measure. First, its value needs to increase as objects are less similar (being more different is represented as being further away from each other). Secondly, measure needs to be symmetrical, meaning that it shouldn't matter whether we measure distance from objects A to B or from B to A. Finally, it needs to be normalized to the $[0, 1]$ interval, which is a convenience that allows easier comparisons between different measures applied on various datasets. Table 1 shows our chosen measures, their notations, range and symmetrical properties.

$P(A)$ is the probability of object A's appearance in a transaction, calculated as a ratio of transactions where the object appears and total number of transactions. Similarly, $P(A, B)$ is the probability of A and B appearing together in a transaction, and $P(A, \bar{B})$ a probability that A will be present and B absent from the transaction.

All chosen measures are conforming to the criteria of being constrained to the $[0, 1]$ interval. However, regarding other distance measure criteria, certain adjustments need to be made. Support measures how often items appear together in transactions, while confidence reflects the probability of an object appearing

Table 1. Chosen measures and their properties

Notation	Name	Formula	Range	Symm?
s	support	$s(A \Rightarrow B) = s(A,B) = P(A,B)$	$0 \ldots 1$	Yes
c	confidence	$c(A \Rightarrow B) = \frac{P(A,B)}{P(A)}$	$0 \ldots 1$	No
ABD	asymmetric binary dissimilarity	$ABD(A,B) = \frac{P(A,\bar{B})+P(\bar{A},B)}{P(A,B)+P(A,\bar{B})+P(\bar{A},B)}$	$0 \ldots 1$	Yes

in a transaction if it is known that another object is present. As such, these measures achieve higher values when the observed pairs of objects are more closely connected. Since we want exactly the opposite, following transformation can be applied:

$$m_x = 1 - x \tag{1}$$

where x represents the chosen measure (in this case support/confidence) and m_x the adjusted measure.

Confidence measure has another problem, which is that it isn't symmetrical, meaning that in general $c(A \Rightarrow B) \neq c(B \Rightarrow A)$. If confidence is to be used as a distance measure, a choice must be made which of the possible two variants will be used as the default one. Following logical reasoning and the guideline in [5] symmetrical variant of the confidence measure could default to a larger of two possible values:

$$c_{sim}(A,B) = max\left\{c(A \Rightarrow B), c(B \Rightarrow A)\right\} \tag{2}$$

The asymmetrical binary dissimilarity measure fits all distance measure requirements so it doesn't need further adjustments, it can be used as a distance measure in its original form. Usage of this measure stems from the fact that nonoccurence of two elementes in a transaction is not as important as their mutual occurence in the same transaction.

The final transformation formulas of discussed measures can be seen in the following subsection, Table 2.

3.2 Integrating Quantitative Attributes in Distance Measure Formulas

Measures discussed in previous subsection offer a good representation of distance and make hierarchical clustering method feasible for transactional data analysis. However, an issue that may arise is the fact that in discussed analysis all observed objects, when perceived by themselves, are essentially considered as being of equal worth. In realistic analysis this would probably not be the case - depending on the goals of the analysis, the analyst or business user will usually use an inherent preference system which will value some objects over others. In retail systems for example, higher priced items or items commonly bought in larger quantities would be valued higher than others because they contribute more to the intended goal of achieving larger profit. For this reason we have decided to look into possible integration of quantitative attributes into developed distance measures, so the gained results may conform more closely to the expected goals of the analysis. We have also put emphasis on the semantics behind chosen quantitative attributes and the way they affect the expected results.

Since retail industry is where transactional data analysis is most commonly required, and which uses most straightforward quantitative attributes, we have focused our efforts on that particular field. However, transformations of data from other fields with specific weights given to transaction elements enable broad usage spectrum of developed measures. In retail industry, common goal is achieving larger profits, and quantitative attributes related to observed transaction items are quantity and price. We will therefore define a notion of normalized turnover of an object. Turnover of an object may be calculated as a multiplication of the price of an object with the average quantity of the object per transaction. This value can further be normalized to the $[0, 1]$ interval so it can be used as a convenient weight or a ponder. We calculate a normalized turnover of an object A with:

$$t(A) = \frac{T(A)}{T_{s.max}} = \frac{P_c(A) \cdot Q_{AVG}(A)}{max_{X \in S} \left\{ P_c(X) \cdot Q_{AVG}(X) \right\}} \tag{3}$$

where T denotes actual average turnover, P_c denotes price, Q_{AVG} average quantity and S a set of observed items that appear in transactions.

We can also define a normalized turnover of pairs of objects in a similar way:

$$t(A, B) = \frac{T(A, B)}{T_{p.max}} = \frac{T(A) + T(B)}{max_{X,Y \in S} \left\{ T(X) + T(Y) \right\}} \tag{4}$$

Support measure reflects how often two objects are bought together. If we multiply the support value with a turnover value, the final measure will represent an approximation of expected contribution to the goal given by objects A and B when viewed as a pair. After applying the transformation 1 to conform to the distance measure requirements we come up with the final measure named *weighted support*, which can be seen in Table 2.

Semantics behind confidence measure reflect how presence of one object affects the probability of another object being present. We have devised two ways of integrating quantitative attributes to this measure. They both rely on the logical assumption that the analyst may prefer learning about the cases when less valuable object results in a more valuable object appearing than vice versa. The first enforces the rule that the less valuable object needs to be the antecedent. The second method is less restrictive and only mandates that the confidence value is multiplied with the normalized turnover of the consequent before defaulting to the larger value. Two final formulas can be seen in Table 2.

Finally, when considering asymmetric binary dissimilarity, similar reasoning already discussed for the support measure can be followed. We may integrate quantitative attributes by multiplying the measure with the inverse turnover

Table 2. Derived distance measures with integrated quantitative parameters

Source measure	Distance formula	Adjusted formula w/ quantitative parameters
support	$1 - s(A, B)$	$1 - s(A, B) \cdot t(A, B)$
confidence	$1 - max\,\{c(A \Rightarrow B), c(B \Rightarrow A)\}$	$t(A) < t(B) : 1 - c(A \Rightarrow B) \cdot t(B)$ $t(A) \geq t(B) : 1 - c(B \Rightarrow A) \cdot t(A)$
		$1 - max\{c(A \Rightarrow B) \cdot t(B), c(B \Rightarrow A) \cdot t(A)\}$
ABD	$ABD(A, B)$	$ABD(A, B) \cdot [1 - t(A, B)]$

value for pairs of objects. Final measure is simply called *weighted asymmetric binary dissimilarity*, with the formula present in Table 2.

4 Implementation and Verification

4.1 Implementation

For verification and testing purposes we have implemented a new custom widget for the Orange data mining tool environment [14] called "Quantitative Attribute Distance" widget. It shares the input/output signals and overall context with the default "Attribute Distance Widget" which makes its usage extremely easy and keeps the learning curve rather low. Figure 1 shows the screenshot of the Orange schema using our widget together with the widget interface and options. By using this widget we were able to put all the developed measures in action and compare their results in tabular and dendrogram (Fig. 2) forms.

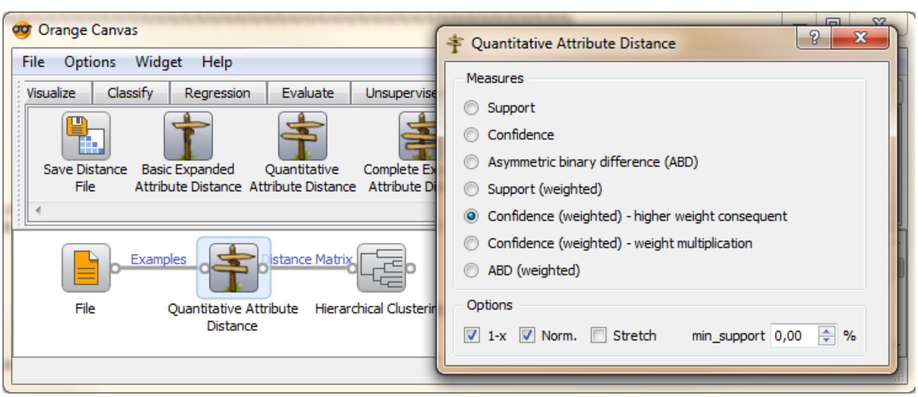


Fig. 1. 'Quantitative Attribute Distance' widget in Orange data mining tool

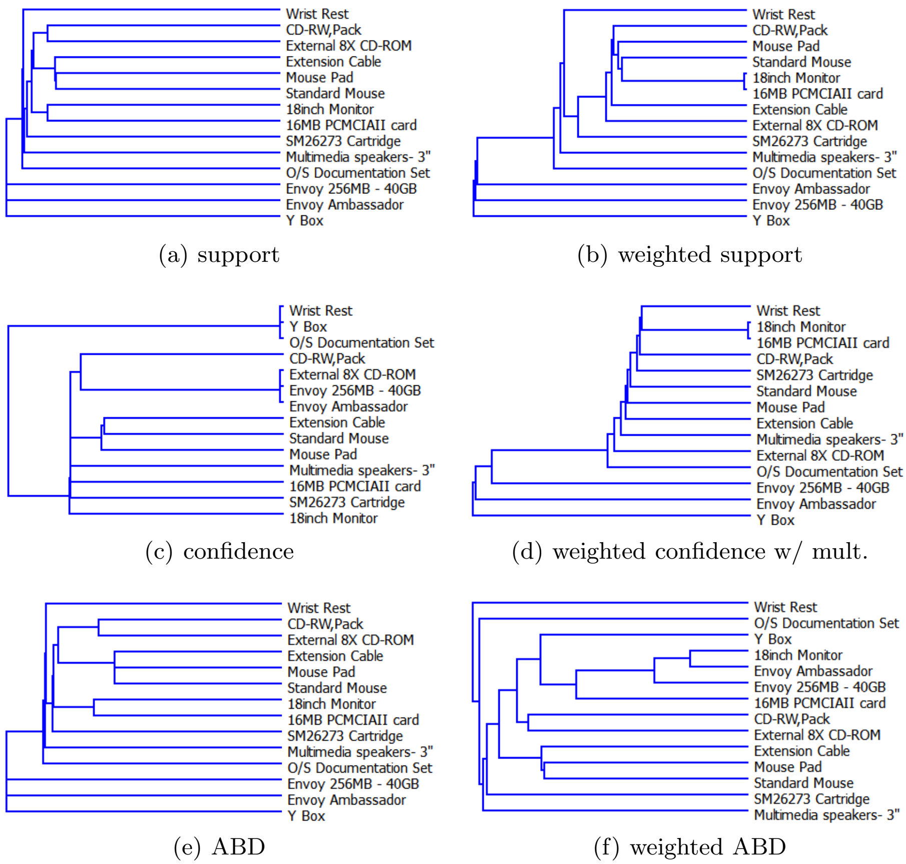


Fig. 2. Resulting dendrograms across all the measures

4.2 Referent Dataset Analysis and Results Evaluation

To adequately test newly developed measures we used two different datasets. 'Computer shop' transactional dataset (available as one of the test datasets bundled with Oracle 11g Database) is a rather small dataset and thus convenient for basic testing. Some earlier research regarding tree structures formations over 'Computer shop' dataset is presented in [15]. Final verification and application of developed measures was made on a large real life retail dataset.

'Computer Shop' dataset contains 940 transactions and 14 transaction elements. Each transaction is a record of a purchase in a shop dealing with electronic equipment. Average number of elements per transaction is 2.98, most frequent element appears in 32% of transactions, and average purchased quantity per item is 1. Prices are in quite a wide range: \$12,55 - \$1565,18. Results with usage of simplest distance measure derived from support reveals relationships presented in Fig. 2(a). As expected, peripherals are bundled together, and so is equipment which fits together logically.

Fig. 2(b) reveals a dendrogram modeled with weighted support using single linkage criterion. Dendrogram reveals that from perspective of element pairs' appearance and their prices most cost effective items are those related to graphics. Further connections to peripheral items are a result of their frequent appearance in transactions which overcame the effect of low prices.

Dendrogram formed with usage of confidence - weight multiplication, average linkage criterion, is presented in Fig. 2(d). It particularly emphasizes situations where appearance of one element in transaction suggests possible turnover achieved by other elements appearing in the same transactions. Confidence is an asymmetric measure, and in most cases when multiplying confidence with consequent's value, relationship where more expensive item is playing consequent role prevails. Experiments show that for 'Computer shop' dataset using confidence without quantitative attributes produced only 35 relationships where cheaper items lead to more expensive ones, and 56 relationships of the opposite direction. This is not what we desire in this context. With prices taken into account situation changes and there are 62 out of 91 relationships which reveal indication that the appearance of a cheaper item indicates possible turnover gain achieved by more expensive item turning up in a transaction. Dendrogram enables easy and quick identification of these items that are most likely to affect one another and bring up profits.

As said before, confidence is not symmetrical and there are still 29 relationships where conditional probability overcomes the effect of the consequent's price. To investigate how these relationships affect final model, alternative confidence measure with higher weight consequent is used. Insight in gained model with usage of this measure reveals only slight changes in connections near to the dendrogram root. Thus, analyst can by using both derived measures with certainty conclude which item could boost up turnover, taking into account conditional probability. In our case appearance of 16MB PCMCIAII card means high value of probable turnover gained by appearance of 18inch Monitor in the same transaction. Of course, analyst should be aware that reasoning about causality cannot be solely based on the results in dendrograms or association rules.

When talking about ABD measure gained model is very similar to model derived by support. When boosting up relationships on bases of weights derived from prices and quantities, most expensive items come to foreground (Fig. 2(f)). This measure is expected to be of more value to the analyst in analysis of datasets where the range of prices of items is not so huge.

The other dataset we analyzed was a real-life retail dataset with 73009 transactions related to 19893 products, which belong to 1368 categories organized in 4 hierarchical layers. Multiple analysis were conducted, out of which one will be briefly discussed. This one concerns the subset of 29 lowest layer categories dealing with cosmetic products. Expected goal of the analysis was insight into relationships between categories with the emphasis on realized turnover.

The transformation and filtering process left 23674 transactions. Each observed category held approximately 60 products, the attributes of which were used to calculate certain aggregate values used to describe the categories. Using guidelines from [13], each category was attributed an average turnover for all the transactions in which at least one product of that category was present. Standard deviation and other statistical measurements affirmed that calculated values were adequate representations of quantitative category attributes ([13]). Interestingly, resulting values of average worth were spread evenly over the manageable interval of [19, 93] without any outliers, which facilitated their comparison.

After the hierarchical clustering method was applied with previously discussed measures, certain interesting conclusions have been reached. As expected, weighted support has clearly identified the most profitable pairs of categories. Both weighted confidence alternatives gave similar results with emphasized relationships between logically connected categories. The clear representation of elements' ordering was especially valuable. Following the ways categories connected and the order of those connections offered concise and beneficial insight in realized turnover and possible ways of increasing it. Certain results were rather unexpected which also showcased the potential worth of realized models.

Experiments with various linkage criteria show stability of some dendrograms and offers some new insights to analysts. When analyst is focused on specific items or categories option of selecting only those in Orange environment speeds up the process of decision making even further.

5 Conclusion

In this paper we present how hierarchical clustering with adequately adjusted distance measures can be used to analyze transactional data. Special emphasis is placed on quantitative attributes associated with transactional elements. Inclusion of quantitative attributes which uphold analyst's goal in distance measures formula enabled generation of dendrograms proven to be more valuable to business analysts.

Hierarchical clustering usage results in completely different models than association rules generation. It relies on establishing distances between pairs of elements and is thus much less resource intensive. Ability to directly employ quantitative attributes and conciseness of results is of great benefit in a decision making process, as long as the analyst is aware of the certain degree of information loss which is always present in concise representations.

Developed measures were implemented within an open source data mining tool and tested on real life datasets. Our research on retail datasets proved that each developed measure provides a new perspective on transactional items' relationships and clients' buying habits. Connections between elements that could bring up turnover and revenue are specially emphasized. Datasets where developed measures could be used are not limited to retail only. Future work includes converting datasets from other domains to transactional data with special consideration given to attributes connected to elements' values. It also includes planned collaboration with domain experts to ensure proper feedback and further improvements of devised measures.

References

1. Han, J., Kamber, M.: Data mining: concepts and techniques. The Morgan Kaufmann series in data management systems. Elsevier (2006)
2. Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, SIGMOD 1993, New York, pp. 207–216 (1993)
3. Padmanabhan, B.: The interestingness paradox in pattern discovery. Journal of Applied Statistics 31(8), 1019–1035 (2004)
4. Piatetsky-Shapiro, G.: Discovery, analysis and presentation of strong rules. In: Knowledge Discovery in Databases, pp. 229–248. AAAI Press (1991)
5. Tan, P.N., Kumar, V., Srivastava, J.: Selecting the right objective measure for association analysis. Information Systems 29, 293–313 (2004)
6. Webb, G.I.: Discovering significant rules. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2006, New York, pp. 434–443 (2006)
7. Webb, G.I.: Self-sufficient itemsets: An approach to screening potentially interesting associations between items. ACM Transactions on Knowledge Discovery From Data 4, 1–20 (2010)
8. Pinjušić, S., Vranić, M., Pintar, D.: Improvement of hierarchical clustering results by refinement of variable types and distance measures. Automatika: Journal for Control, Measurement, Electronics, Computing and Communications 52(4), 353–364 (2011)
9. Vranić, M.: Designing concise representation of correlations among elements in transactional data. PhD thesis, FER, Zagreb, Croatia (2011)
10. Vranić, M., Pintar, D., Gamberger, D.: Adapting hierarchical clustering distance measures for improved presentation of relationships between transaction elements. Journal of Information and Organizational Sciences 36(1) (in press, 2012)
11. Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. SIGMOD Rec. 25, 1–12 (1996)
12. Ruckert, U., Richter, L., Kramer, S.: Quantitative association rules based on halfspaces: An optimization approach. In: Proceedings of the Fourth IEEE International Conference on Data Mining, ICDM 2004, pp. 507–510. IEEE Computer Society, Washington DC (2004)
13. Aumann, Y., Lindel, Y.: A statistical theory for quantitative association rules. Journal of Intelligent Information Systems, 261–270 (1999)
14. Demšar, J., Zupan, B., Leban, G., Curk, T.: Orange: From Experimental Machine Learning to Interactive Data Mining. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) PKDD 2004. LNCS (LNAI), vol. 3202, pp. 537–539. Springer, Heidelberg (2004)
15. Vranić, M., Pintar, D., Skočir, Z.: Generation and analysis of tree structures based on association rules and hierarchical clustering. In: Proceedings of the 2010 Fifth International Multi-conference on Computing in the Global Information Technology, ICCGI 2010, pp. 48–53. IEEE Computer Society, Washington DC (2010)

The MARS - A Multi-Agent Recommendation System for Games on Mobile Phones

Pavle Skocir, Luka Marusic, Marinko Marusic, and Ana Petric

University of Zagreb
Faculty of Electrical Engineering and Computing
Unska 3, HR-10000, Zagreb, Croatia
{pavle.skocir,luka.marusic,marinko.marusic,ana.petric}@fer.hr

Abstract. In order to achieve flow (i.e. complete focus on playing followed by a high level of enjoyment) and increase player retention (i.e. keep a user playing a game longer and more often) it is important that difficulty of the game that a user is playing matches her/his skills. Due to a large amount of different games which are available to users, it is not easy for them to find games which best suit their skills and abilities. In this paper we propose a recommendation algorithm based on the information gathered from users' interaction with a game. We use that information to model users' success and progress in the game as well as motivation for playing. Besides, the proposed algorithm also takes into account user preferences, mobile phone characteristics and game related information which is gathered from users once the game is available on the market. Before enough information is gathered from users, the algorithm uses the information gathered during the game development phase and acquired from game developers and testers. In the implemented multi-agent system, after a user finishes playing a game, she/he receives a notification with a list of games which best suit her/his skills and preferences.

Keywords: computer and video games, user experience, recommendation system, multi-agent system.

1 Introduction

The amount of content which various service providers offer to their users is rapidly rising every day. Games, as a very popular content type, are no exception to that rule. According to Entertainment Software Association's (ESA[1]) annual report [1] in almost three quarters of American households people play computer or video games and 55% of them play those games on their mobile phones and/or handheld devices. Since people (purchase and) play those games to have fun, entertaining the player and achieving player enjoyment is considered to be the most important goal for computer and video games.

[1] http://www.theesa.com/

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 104–113, 2012.

Different methods for evaluating gameplay experience are used in all phases of the game development process with the purpose of improving design of games and various aspects of gameplay experience [2,10]. Some of those methods include self-reporting (e.g., interviews, questionnaires), measuring psychological state of the player (e.g., hart rate, galvanic skin response) [15] and behaviour analysis from game metrics [21].

Two related concepts which are often used to describe and model gameplay experience are flow [6,20] and immersion [3,8]. Flow can be defined as the feeling of complete focus in the game followed by a high level of enjoyment [5] while immersion is used to describe the degree of engagement or involvement in the game [3]. In their own way, both concepts are closely related to the challenge a game represents to the user. The balance between a challenge of successfully completing a certain aspect of the game (i.e. game task or level) and user's skills and abilities to do so is considered to be an important precursor to flow [5,20].

Another measure for successfulness of the game is player retention, i.e. the ability of a game to keep a user playing the game for longer periods of time and/or returning to play a game later [17]. Research has shown that it is necessary that the game quickly adapts to the user (i.e. finds a balance between her/his skills and game difficulty) in order to improve player retention [22].

In our research, we use those findings to model a multi-agent recommendation system where agents monitor users' in-game behaviour and analyse their progress in a certain game. The results of the analysis are used to recommend games which best suit users' skills and abilities in order to longer maintain users' flow and increase the time users spend playing a certain game.

The paper is organized as follows. Section 2 gives an overview of related work on recommendation systems for other content types, while Section 3 describes the model of the proposed recommendation system and the architecture of the implemented multi-agent system. Section 4 presents the recommendation algorithms which take into account modelled user experience parameters as well as user preferences. Section 5 concludes the paper and gives outline for future work.

2 Related Work

Liang et al. [13] showed that the problem of content overload can be reduced using personalized services thus increasing users' satisfaction and offering content which best suits user preferences. Therefore, if service providers recommend content that meets users' interest, users save time and effort because they do not have to look for potentially interesting content on their own. There exist various content recommendation systems for different content types (e.g., video, music, books) and here we will briefly present the main concepts from a few familiar ones.

One of the well known recommendation systems is the one used by YouTube[2] [7]. When forming recommendations it considers data collected from recent user actions (e.g., watched, liked video) and content data (e.g., raw video streams as

[2] http://www.youtube.com

well as video metadata) including periodically calculated relatedness score for each pair of videos depending on how often were those videos co-watched within sessions. Recommended videos are scored and ranked in dependence of video quality, user specificity and diversification criteria.

Another known recommendation system is the one used by Amazon[3]. It uses item-to-item collaborative filtering method in its online store [14]. Unlike most recommendation systems which organize similar users into groups and offer them products that other users from their group purchased/viewed/positively rated, Amazon's algorithm finds items that are similar to the ones that the user already purchased or rated. The found items are aggregated and the most popular or correlated ones are recommended to the user. The table of similar items is created offline and it is used by the algorithm's online component to create recommendations for users.

There also exist systems that use several different means for recommending products to users, like the system proposed in [4] that uses fuzzy logic and data mining techniques to create recommendations. Based on its domain expert knowledge and information obtained from a user, the system uses multi-attribute decision making method to calculate the optimality of each product and recommends the best ranked ones. In recommendation systems, software agents can be used for collecting information about users' content consumption by monitoring their actions and modelling their preferences [12] or for learning users' preferences from their recent actions and giving personalized suggestions based on those preferences [16].

3 A Multi-Agent System for Game Provisioning

This section describes the Multi-Agent Recommendation System (MARS) designed to recommend games for mobile phones to users based on their skills and abilities demonstrated in previous games they played. The recommendation model and the architecture of the implemented multi-agent system are presented.

3.1 The Model of the Recommendation System

Fig. 1 shows the recommendation model for games which is based upon four main processes: playing games, analysing game consumption data, recommending and provisioning games. While a user plays a game, an agent monitors her/his activities on her/his mobile phone which concern the game downloaded from the service provider. Collected information is stored in a database and analysed afterwards. The results of the analysis are used to recommend new games to users as well as to enhance the decision making process for purchasing content distribution rights for new games from various content providers [18]. Once the content provider delivers games to the service provider, new recommendation lists are created and games are recommended to users in correspondence with users' skills and preferences. If a user likes the recommended game she/he downloads it and plays it on her/his mobile phone.

[3] http://www.amazon.com

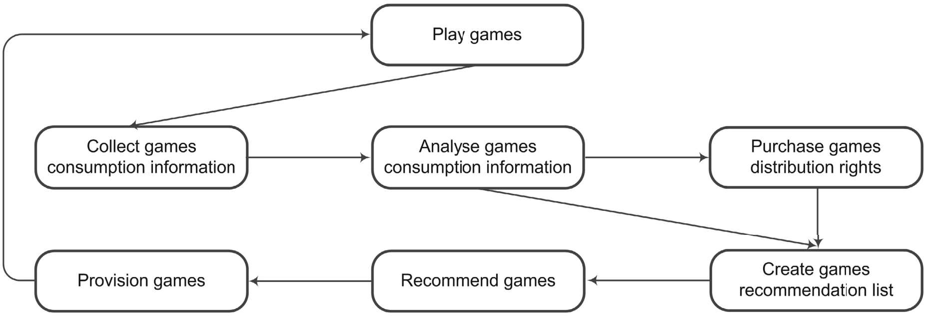


Fig. 1. The games recommendation model

3.2 The MARS Architecture

Agents in the MARS system enable trading on two different electronic markets (e-markets): business-to-customer (B2C) and business-to-business (B2B) content e-market. As shown in Fig. 2, there are four types of agents in the MARS system: Service Provider Agent (SPA), Database Agent (DA), Content Provider Agent (CPA) and User Agent (UA). The SPA represents a network operator which acts as a service provider on the content e-market while the DA is in charge of SPA's database. The CPAs represent game publishing companies which publish (and produce) games and act as content providers on the B2B content e-market. On the B2C content e-market, the SPA provides game-based services to users which are represented by their UAs. The MARS system enables automated trading on the B2B and B2C content e-markets. Based on users' skills and preferences, the SPA recommends suitable games to its users while, based on the gameplay statistics, it purchases content distribution rights from CPAs for new similar games.

Digital games which a user purchased from the service provider as well as the UA are located on her/his mobile phone. The UA monitors and analyses user's interaction with the game and determines the game category and difficulty level which best correspond with her/his skills and preferences. Afterwards, the UA sends a request to the SPA asking for a new game recommendation and waits for a response.

Upon receiving a request from the UA, the SPA carries out recommendations based on the game category and difficulty level specified in the UA's request but it also takes into account information specified in the user profile and the profile of the user's mobile phone. The SPA sends a new game recommendation to the UA which displays game information to the user. Additionally, the SPA forwards received information about game consumption (e.g. how often and how long did a user play a game) to the DA for the database update. Periodically, the SPA contacts the CPA and negotiates the purchase of content distribution rights for new games [19].

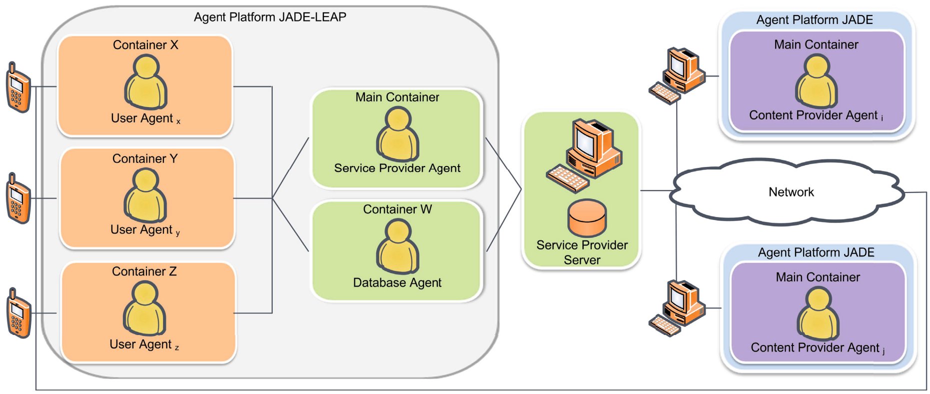


Fig. 2. The architecture of the MARS system

4 Gameplay Data and User Preferences Analysis for Game Recommendation

In this section, parameters used to model user's skills and gameplay experience are described and afterwards the algorithms for processing data collected during user-game interaction are explained in detail.

4.1 The Parameters for Analysing User's Skills

According to Komulainen et al. [11], gameplay experience consists of: cognition, motivation, emotion and focused attention. Since cognition is difficult to analyse using existing research techniques and emotions are analysed using questionnaires and measuring psychological state of the users, we observed the remaining two elements. In this paper, we propose models of motivation as well as of focused attention that is represented with user's success and progress in the game, as the parameters used for recommending trivia games.

For the estimation of user's success and progress, following parameters are needed: $\bar{t_{ij}}$ - average time that the user i needed to successfully complete game level j; $\bar{t_j}$ - average time needed to successfully complete game level j for all players $i \in [1, n]$; and precision in mastering the game (i.e. was the given solution true or false or was the given task successfully performed). The parameter t_{ij} is defined as:

$$t_{ij} = t_{ijs} + \sum_{f=1}^{p} t_{ijf}, \tag{1}$$

where t_{ijs} represents the playtime of the level j when the user i successfully completed it, while the t_{ijf} represents the playtime of the level j when the user i failed to successfully complete the level assuming that p is the total number of failed attempts. If the user i played the game more than once (e.g., k times) than the average time she/he needed to complete the level j is calculated as:

$$\bar{t_{ij}} = \frac{1}{k}\sum_{l=1}^{k} t_{ij}^{l} = \frac{1}{k}\sum_{l=1}^{k}(t_{ijs}^{l} + \sum_{f=1}^{p} t_{ijf}^{l}), \tag{2}$$

where t_{ij}^l, t_{ijs}^l and t_{ijf}^l denote playtimes measured during user's l^{th} running of the game. Average time $\bar{t_j}$ that a total of n users which played the game need to complete the level j is calculated as:

$$\bar{t_j} = \frac{1}{n}\sum_{i=1}^{n} \bar{t_{ij}} = \frac{1}{n}\frac{1}{k}\sum_{i=1}^{n}\sum_{l=1}^{k}(t_{ijs}^{l} + \sum_{f=1}^{p} t_{ijf}^{l}). \tag{3}$$

User's success is compared with the information how successfully other users play that game level. The gameplay metrics of all users are stored in the service provider's database and their average playtimes are calculated using Equations (1), (2) and (3). The UA receives information with those average results from the SPA. If the user i needs significantly less time to complete the game level j than other users do (i.e. $\bar{t_{ij}} \ll \bar{t_j}$), then she/he is considered to be more successful than other users. The progress is calculated from the information about user's earlier success results in that game. If user i's recent results are better than the past ones (i.e. the user i needs less time to complete the game level j than earlier; $t_{ij}^l < \bar{t_{ij}}$), then it is considered that the user i progresses.

We also model user's motivation which can be defined as a psychological state which reflects user's desire to play a game [9]. The parameters needed for the estimation of user's motivation are: t_i - the time that the user i plays a certain game; and $\bar{t}$ - average time all users spent playing that game. The parameter t_i is defined as:

$$t_i = \sum_{j=1}^{m}(t_{ijs} + \sum_{f=1}^{p} t_{ijf}), \tag{4}$$

where m is the highest level that the user i reached while playing the game. Average time $\bar{t}$ that a total of n users spent playing a game is calculated as:

$$\bar{t} = \frac{1}{n}\sum_{i=1}^{n}\bar{t_i} = \frac{1}{n}\frac{1}{k_i}\sum_{i=1}^{n}\sum_{l=1}^{k_i} t_i^l = \frac{1}{n}\frac{1}{k_i}\sum_{i=1}^{n}\sum_{l=1}^{k_i}\sum_{j=1}^{m_l}(t_{ijs}^{l} + \sum_{f=1}^{p} t_{ijf}^{l}), \tag{5}$$

where k_i denotes how many times has the user i played the game while m_l is the highest level that the user i reached during its l^{th} running of the game. If the user i spends more time playing a certain game than the average user does (i.e. $\bar{t_i} > \bar{t}$), than we assume she/he is highly motivated and would like to play other games of the same category.

4.2 The Recommendation Algorithms

As shown in Algorithm 1, the UA compares user's current gameplay results, her/his historic results and average results of all users for that game which are obtained from the SPA. While the user i plays a game, the UA collects the following information about the game: the level user played (i.e. j), the time she/he spent playing the level j (i.e. t_{ij}), the total time she/he spent playing the game (i.e. t_i) and the result she/he achieved (i.e. variable *result* in Algorithm 1).

Since games are tested and evaluated by experts and regular players several times in different phases of their development [2], content providers (i.e. game publishing companies) provide information about the estimated average time needed to complete game tasks. Before it collects sufficient gameplay data from a certain game that is used to calculate the average time needed to complete a game task (i.e. $\bar{t_j}$), the SPA uses the estimates received from the CPA. The UA measures the time (i.e. t_{ij}) the user i needs to successfully complete a task (i.e. level j) and compares it with the average time of other players (i.e. $\bar{t_j}$) in order to determine user's success. Together with the number of times (i.e. p) the user i unsuccessfully attempted to solve a task those data are used to keep track of user's progress in the game.

Content providers also estimate the average total time a user will spend playing a game (i.e. $\bar{t}$). The UA keeps track how many times the user i played a game (i.e. k) and how long did each of those play sessions last (i.e. t_{ij}). From that information the UA calculates the time the user i played a game (i.e. t_i) and uses it to determine user's motivation by comparing it with the information about average gameplay duration (i.e. $\bar{t}$).

Algorithm 1: Determining the category and difficulty of a new game

$playedCategory$ = category of the currently played game;
$playedDifficulty$ = difficulty of the currently played game;
$result$ = denotes was the user's solution of a task successful;
switch *result*, $\bar{t_{ij}}$, $\bar{t_j}$, t_i, $\bar{t}$ **do**
  case $result == false$ && $t_i > \bar{t}$
    $newCategory = playedCategory$;
    $newDifficulty = playedDifficulty - 1$;
  case $result == false$ && $t_i < \bar{t}$
    $newCategory = findAnotherCategory()$;
    $newDifficulty = playedDifficulty$;
  case $result == true$ && $\bar{t_{ij}} < \bar{t_j}$ && $t_i > \bar{t}$
    $newCategory = playedCategory$;
    $newDifficulty = playedDifficulty$;
  case $result == true$ && $\bar{t_{ij}} < \bar{t_j}$ && $t_i < \bar{t}$
    $newCategory = playedCategory$;
    $newDifficulty = playedDifficulty + 1$;
  case $result == true$ && $\bar{t_{ij}} > \bar{t_j}$ && $t_i > \bar{t}$
    $newCategory = playedCategory$;
    $newDifficulty = playedDifficulty - 1$;
  case $result == true$ && $\bar{t_{ij}} > \bar{t_j}$ && $t_i < \bar{t}$
    $newCategory = findAnotherCategory()$;
    $newDifficulty = playedDifficulty$;
endsw

If the user i plays a game longer than the average time calculated by the content provider (i.e. $\bar{t}$), it is assumed that the user is highly motivated and a game from the same category is recommended, otherwise she/he is poorly motivated and a game from another category is recommended. If the user i is more successful than an average player, a more difficult game of the same category is recommended in order to maintain user's flow [20] and improve player retention [22]. To a highly motivated user which demonstrates satisfactory progress, a similar game from the same category and with similar difficulty is recommended. However, if user's progress is satisfactory but the motivation is poor a more difficult game is recommended to avoid boredom since it is assumed that this game is not challenging enough for this user [6]. We assume that the game is to difficult for the user if she/he is motivated but does not show progress in the current game so an easier game is recommended [5]. In case the motivation is also lacking, a game from another category is recommended since it is considered that the user is not interested in similar games.

Based on this analysis of user's success, progress and motivation, the UA determines the category and difficulty of a new game that is believed to be suitable for its user. Those parameters are sent to the SPA which, as shown in Algorithm 2, finds the most appropriate games for the given user by taking into account received parameters and user profile (e.g., age group - children, teenagers, students, adults, seniors) and her/his mobile device profile (e.g., operating system, screen size, available memory). If games of desired category and difficulty cannot be found, a random game for user's mobile phone is recommended.

Algorithm 2: Selecting a game to be recommended

```
phoneID = findPhone (userID);
games = findGames (newCategory, newDifficulty, phoneID);
if games.size > 0 then
    games2 = findNotDownloadedGames (games, userID);
    if games2.size > 0 then
        return randomGame (games2);
    else
        return randomGame (games);
    end
else
    games3 = findGames (otherCategory, newDifficulty, phoneID);
    games4 = findNotDownloadedGames (games3, userID);
    return randomGame (games4);
end
```

5 Conclusion and Future Work

We presented the Multi-Agent Recommendation System (MARS) for games on mobile phones which enables game recommendation based on user's consumption information, game related information and user preferences (i.e. user profiles and mobile phone profiles). Software agents were chosen as representatives of stakeholders in the system because they can autonomously collect data for their owners, analyse that data and recommend business decisions which will maximize service provider's profit and enhance user experience for existing as well as for new services. We proposed a recommendation algorithm in which decisions are made based on user's motivation, success and in-game progress which were modelled by using the temporal dimension of games. The recommended games match user's skills and preferences and thus maintain user flow experience and improve player retention.

The implemented B2C content e-market enables game downloading, collecting and processing of gameplay data and recommending games. For future work we plan to fully integrate the MARS system with the B2B content e-market for trading with content distribution rights [19] and introduce the results of the gameplay data analysis as one of the parameters taken into account in the decision making process used for purchasing content distribution rights [18].

Acknowledgments. The authors acknowledge the support of research project "Content Delivery and Mobility of Users and Services in New Generation Networks" (036-0362027-1639), funded by the Ministry of Science, Education and Sports of the Republic of Croatia.

References

1. Essential Facts about the Computer and Videogame Industry, 2011 Sales, Demographic and Usage Data. Entertainment Software Association, Washington, DC, USA (2011), http://www.theesa.com/facts/pdfs/ESA_EF_2011.pdf
2. Bernhaupt, R., Linard, N.: User Experience Evaluation for Multimodal Interaction in Games. In: Raymaekers, C., Coninx, K., González-Calleros, J.M. (eds.) Design and Engineering of Game-like Virtual and Multimodal Environments, pp. 19–22 (2010)
3. Brown, E., Cairns, P.: A grounded investigation of game immersion. In: CHI 2004 Extended Abstracts on Human Factors in Computing Systems, pp. 1297–1300. ACM, New York (2004)
4. Cao, Y., Li, Y.: An intelligent fuzzy-based recommendation system for consumer electronic products. Expert Systems with Applications 33(1), 230–240 (2007)
5. Chen, J.: Flow in games (and everything else). Communications of the ACM 50, 31–34 (2007)
6. Cowley, B., Charles, D., Black, M., Hickey, R.: Toward an understanding of flow in video games. Computers in Entertainment 6, 20:1–20:27 (2008)
7. Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., Sampath, D.: The YouTube video recommendation system. In: Proceedings of the Fourth ACM Conference on Recommender Systems, pp. 293–296. ACM, New York (2010)
8. Ermi, L., Mäyrä, F.: Fundamental components of the gameplay experience: Analysing immersion. In: de Castell, S., Jenson, J. (eds.) Changing Views: Worlds in Play, pp. 15–27. University of Vancouver (2005)

9. Fernandez, A.: Fun experience with digital games: a model proposition. In: Olli, L., Hanna, W., Amyris, F. (eds.) Extending Experiences. Structure, Analysis and Design of Computer Game Player Experience, pp. 181–190. Lapland University Press (2008)
10. Kim, J.H., Gunn, D.V., Schuh, E., Phillips, B., Pagulayan, R.J., Wixon, D.: Tracking real-time user experience (TRUE): a comprehensive instrumentation solution for complex systems. In: Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems, CHI 2008, pp. 443–452. ACM, New York (2008)
11. Komulainen, J., Takatalo, J., Lehtonen, M., Nyman, G.: Psychologically structured approach to user experience in games. In: Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges, pp. 487–490. ACM, New York (2008)
12. Lee, W.P., Liu, C.H., Lu, C.C.: Intelligent agent-based systems for personalized recommendations in internet commerce. Expert Systems with Applications 22(4), 275–284 (2002)
13. Liang, T.P., Lai, H.J., Ku, Y.C.: Personalized Content Recommendation and User Satisfaction: Theoretical Synthesis and Empirical Findings. Journal of Management Information Systems 23, 45–70 (2007)
14. Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing 7, 76–80 (2003)
15. Mandryk, R.L., Inkpen, K.M., Calvert, T.W.: Using psychophysiological techniques to measure user experience with entertainment technologies. Behaviour & Information Technology 25(2), 141–158 (2006)
16. Miao, C., Yang, Q., Fang, H., Goh, A.: A cognitive approach for agent-based personalized recommendation. Knowledge-Based Systems 20(4), 397–405 (2007)
17. Nacke, L., Drachen, A.: Towards a framework of player experience research. In: Proceedings of the Second International Workshop on Evaluating Player Experience in Games at FDG 2011, Bordeaux, France (2011)
18. Petric, A., Jezic, G.: Multi-attribute Auction Model for Agent-Based Content Trading in Telecom Markets. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010, Part I. LNCS, vol. 6276, pp. 261–270. Springer, Heidelberg (2010)
19. Petric, A., Jezic, G.: A Multi-Agent System for Game Trading on the B2B Electronic Market. In: O'Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS, vol. 6682, pp. 332–341. Springer, Heidelberg (2011)
20. Sweetser, P., Wyeth, P.: Gameflow: a model for evaluating player enjoyment in games. Computers in Entertainment 3(3), 3–3 (2005)
21. Thurau, C., Drachen, A.: Introducing archetypal analysis for player classification in games. In: 2nd International Workshop on Evaluating Player Experience in Games (epex 2011) (2011)
22. Weber, B.G., Mateas, M., Jhala, A.: Using data mining to model player experience. In: FDG Workshop on Evaluating Player Experience in Games. ACM, Bordeaux (2011)

Informativeness of Inflective Noun Bigrams in Croatian

Damir Jurić, Marko Banek, and Šandor Dembitz

University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, HR-10000 Zagreb, Croatia
{damir.juric,marko.banek,sandor.dembitz}@fer.hr

Abstract. A feature of Croatian and other Slavic languages is a rich inflection system, which does not exist in English and other languages that traditionally dominate the scientific focus of computational linguistics. In this paper we present the results of the experiments conducted on the corpus of the Croatian online spellchecker Hascheck, which point to using non-nominative cases for discovering collocations between two nouns, specifically the first name and the family name of a person. We analyzed the frequencies and conditional probabilities of the morphemes corresponding to Croatian cases and quantified the level of attraction between two words using the normalized pointwise mutual information measure. Two components of a personal name are more likely to co-occur in any of the non-nominative cases than in nominative. Furthermore, given a component of a personal name, the conditional probability that it is accompanied with the other component of the name are higher for the genitive/accusative and instrumental case than for nominative.

Keywords: collocations, declension, named entity recognition, semantics, language technologies.

1 Introduction

Multiword expressions (MWEs) are lexical items consisting of two or more orthographic words, whose properties are not predictable from the properties of the individual orthographic words [16]. MWEs are a very heterogeneous class of constructions, varying from proper names (e.g. *Bill Clinton*, *New York*) to idioms (e.g. *kick the bucket* [15, 16], meaning *die*). MWEs make a significant portion of language corpora and dictionary entries (according to [15], 41% of WordNet [6] entries are MWEs). Detection of MWEs and their meaning is a major issue in several different areas of computational linguistics such as machine translation and spellchecking or predictive typing.

Hascheck (Hrvatski Akademski Spelling Checker, i.e. Croatian Academic Spellchecker) [4, 7] is an online spellchecker for Croatian able to learn words from the texts it receives. Its corpus contained 564,065,881 words on October 31, 2011. The online interface of the spellchecker contains a corrector tool, which offers replacements for errors of edit distance 1 or 2. The edit distance formula [8, 10]

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 114–123, 2012.

calculates the number of insertions, deletions, substitutions or transpositions needed to turn one word (the wrongly spelled one) into another (a correctly spelled one). The correction candidate words of the same edit distance are further sorted according to a weighting factor that includes the frequency of the word in the Hascheck corpus, the keyboard layout and an error classification system conforming to the phonetic characteristics of Croatian. However, the edit distance formula works independently for each word and therefore neglects the fact that the wrongly spelled word may also be a part of an MWE.

The first step for making MWE detection a functionality of the corrector tool is to create a database of named entities (which includes proper names) and then analyze not only the wrongly spelled word but also its surrounding context. Nevertheless, the very fact that the quality of the corrector tool recommendation improved significantly after including the phonetic characteristics of the language into the calculation process suggests that a thorough statistical analysis of Croatian MWEs and MWE candidates within the spellchecker corpus should be performed as well. Each class of MWEs (proper names, idioms, etc.) has properties that are not comparable to other MWE classes and should be analyzed separately.

In this paper we analyze the bigrams of the Hascheck corpus (two words appearing consecutively) and the probabilities of finding two-word proper names. In particular, we examine the collocation probability features arising from the fact that Croatian is a highly-inflective language with a seven-case declension system. The contribution of the paper is to show that for proper name MWEs an MWE particle in a non-nominative case is more likely to form a MWE bigram (i.e. to be immediately followed or immediately preceded by another MWE particle) than an MWE particle in the nominative case. We applied the normalized pointwise mutual information formula [2] to quantify the collocation between the bigram particles. The results of the Pearson's chi-square test reject the possibility that higher collocation for non-nominative cases results from randomness within the error margin.

The paper is structured as follows. Section 2 gives an overview of the related work. The rules for personal name declension in Croatian and their influence on building the corpus for the experiments are explained in Section 3. Section 4 presents the experiments and discusses their results. Conclusions are drawn in Section 5.

2 Related Work

Tools for named entity recognition (NER) are primarily focused on recognizing persons, locations and organizations within a text [12]. NER software has been constructed for different languages, including highly-inflective Slavic languages: Polish [13], Russian [14] and Serbian [9]. The Polish NER tool [13] combines a finite-state machine with a set of heuristic rules. The NER tools for Russian [14] and Serbian [9] are provided with the list of most common first names and family names extracted from phone directory books. The lemmatization modules are able to recognize the number, gender and case of nouns and adjectives in the sentences, as well as the conjugational aspects of the accompanying verbs related to the number and gender of nouns. Interestingly, none of the mentioned tools performs a statistical analysis of a language corpus, which would in our opinion most surely contribute to their better performance (a detailed explanation will be given in Section 3.1).

A variety of measures have been proposed in the literature in order to quantify collocation i.e. the fact that certain words have a tendency to *occur near each other* in a natural language: e.g *cow* and *milk*, *ring* and *bell* or *Harry* and *Potter* (the first two examples are given in [1], the third one is from our research related to this paper). There are three basic approaches to define what *being near each other* means [5]: *surface co-occurrence* (two words co-occur if they appear within a certain distance, measured by the number of intervening word tokens, regardless of the punctuation), *textual co-occurrence* (two words co-occur if they appear in the same textual unit, such as one sentence or one document) and *syntactic co-occurrence* (in this, most restrictive approach two words co-occur if there is a direct syntactic relationship between them, e.g. a verb and its object: *he bought a book* or a noun modified by a prenominal adjective: *The clever boy* wore *a white shirt*). Collocation measures relate the *observed frequency* of the target word pair in the corpus with its *expected frequency* (calculated from the frequencies of each of the two words, as if they co-occured randomly) [5]. An extensive survey of measures, of which the most widely used are *MI* (mutual information) and its different variations, *z-score*, *t-score*, *simple logarithmic likelihood*, *logarithmic likelihood*, *odds-ratio* and *chi-square score*, is given in [5].

3 Hascheck Bigram Database and Declension of Personal Names

As mentioned in Section 1, the corpus of Hascheck contained 564,065,881 words on October 31, 2011. A database of bigrams was extracted from the corpus, containing 46,936,306 different bigrams with a total occurrence of 448,397,465 word pairs separated by blank or dash (in the case of semi-compounds).

The Croatian language belongs to the group of Slavic Indo-European languages. There is rich inflectional system: nouns, pronouns, adjectives and numbers are declined into seven cases (nominative, genitive, dative, accusative, vocative, locative and instrumental), while verbs conjugate for persons and tense. When bigrams consist of two nouns, the nouns may share the number, gender and case: e.g. nominative: *predsjednik Clinton*; genitive: *predsjednika Clintona*, meaning *President Clinton* (in this example the nouns share the male gender). On the other hand, one noun may inflect while the other keeps its non-nominative case: e.g. nominative+genitive: *Predsjednica Republike*; genitive+genitive: *Predsjednice Republike*, meaning a *(female) President (of the) Republic* (accidentally, both nouns are of the female gender). A lemmatization tool for named entity recognition (as mentioned in Section 2), could be enhanced with a search engine over the Hascheck bigram database. Given the bigram *Predsjednica Republike*, the search shows that the genitive *Republike* is accompanied with different cases of *predsjednica*, while there is no nominative-nominative (*Predsjednica Republika*) or instrumental-instrumental bigram (*Predsjednicom Republikom)*, except for spell errors.

When two-word expressions consist of a modifying prenominal adjective and a noun, the two words share the case, the gender and the number (e.g. nominative: *sumporna kiselina*; genitive: *sumporne kiseline*, meaning *sulfuric acid*).

3.1 Declension of Personal Names in Croatian

Personal names in the Western culture typically consist of a first name and a family name. In Croatian, personal names are declined, with the same rules applied both to Croatian and foreign names. Regardless of the actual person's gender, names are declined according to two patterns, which philologists name according to their genitive suffix: the "*a*" pattern (typical for common nouns of the male gender) or the "*e*" pattern (typical for common nouns of the female gender and mostly having the nominative suffix *–a*); *Davor Šuker* and *Bill Clinton* have both their first and family names declined according to the "*a*" pattern (Table 1), Nikola Tesla has both of them declined according to the "*e*" pattern, while for *Barack Obama* or *Diego Maradona* the "*a*" pattern is applied for the first and "*e*" for the family name. The second basic rule states that women's names are not declined at all unless they end with an *–a* (and hence adopt the "*e*" pattern). Thus, *Angela Merkel* has only her first name declined (Table 2), *Michelle Obama* only her family name, while for *Christina Aguilera* both names are declined. *Hillary Clinton*'s first and family name are both kept unchanged.

Table 1. Declension patterns for men's names. The vocative case is not shown

	"a"+"a"	"a"+"e"	"e"+"a"	"e"+"e"
N	Bill Clinton	Barack Obam*a*	Luk*a* Modrić	Nikol*a* Tesl*a*
G	Bill**a** Clinton**a**	Barack**a** Obam**e**	Luk**e** Modrić**a**	Nikol**e** Tesl**e**
D	Billu Clintonu	Baracku Obami	Luki Modriću	Nikoli Tesli
A	Billa Clintona	Baracka Obamu	Luku Modrića	Nikolu Teslu
L	Billu Clintonu	Baracku Obami	Luki Modriću	Nikoli Tesli
I	Billom Clintonom	Barackom Obamom	Lukom Modrićem	Nikolom Teslom

Table 2. Declension patterns for women's names. The vocative case is not shown

	NONE+NONE	NONE +"e"	"e"+ NONE	"e"+"e"
N	Hillary Clinton	Michelle Obam*a*	Angel*a* Merkel	Christin*a* Aguiler*a*
G	Hillary Clinton	Michelle Obam**e**	Angel**e** Merkel	Christin**e** Aguiler**e**
D	Hillary Clinton	Michelle Obami	Angeli Merkel	Christini Aguileri
A	Hillary Clinton	Michelle Obamu	Angelu Merkel	Christini Aguileri
L	Hillary Clinton	Michelle Obami	Angeli Merkel	Christini Aguileri
I	Hillary Clinton	Michelle Obamom	Angelom Merkel	Christinom Aguilerom

3.2 Building the Experiment Corpus

The goal of our research is to check the bigram database of Hascheck in order to determine whether a first name of a person in a non-nominative case will be followed by the family name of that person in the same case more or less frequently than a first name in nominative by the person's family name in nominative. We also check the other direction: is a person's family name more likely to be preceded by the first name in a non-nominative case or in nominative? First, we exclude the vocative case from our experiments since it is by far the least frequent case.

Obviously, the experiment cannot be performed if at least one of the names is not declined. This excludes three of the four pattern combinations for female persons (leaving only the *"e"+"e"* pattern combination as in case of *Christina Aguilera*). Another problem is the fact that genitive and accusative are equal for male names with the "*a*" pattern, but not for male and female names with the "*e*" pattern. The word *Baracka* (see Table 1) can be followed both by *Obame* (genitive) and *Obamu* (accusative), while the association in the opposite direction (*Obame→Baracka*; *Obamu→Baracka*) is unambiguous. Considering the fact that we perform our research in both directions (the first name depending on the family name and the family name depending on the first name), such an asymmetry is not desirable. Thus, men whose first name and family name is a combination of the "*a*" and the "*e*" pattern (*Barack Obama*, *Luka Modrić*) are also excluded.

The two remaining groups are: (1) men whose names follow a double "*a*" pattern (*Bill Clinton*), where declension produces four different morphemes (N, G/A, D/L, I), and (2) men and women whose names adopt a double "*e*" pattern (*Nikola Tesla*, *Christina Aguilera*), producing five morphemes (N, G, D/L, A, I). Since the first group is much more frequent than the second one, we limit the experiment exclusively to the first group.

We handpicked 100 famous men (politicians, sportsmen, actors, musicians and writers) whose name produces at least 10 million hits on Google ™ search engine. One of them is an imaginary person (*Harry Potter*), while all the rest are real. Most of them are living (active, like *Justin Bieber* or retired, like *Michael Jordan*) or recently deceased (*Michael Jackson*). Only *George Washington* and *Abraham Lincoln* were not born in the 20th century. Names pointing to more than a single famous man (e.g. *George Bush*) were not included.

4 Experiments

In the initial step of the experiments, we assigned each of the four morpheme pairs (nominative, genitive/accusative, dative/locative, instrumental) of each of the 100 handpicked male persons' names with three occurrence frequencies in the Hascheck bigram database: the frequency f_{AB} of bigram occurrence where the person's first name is followed by the family name, e.g. (*Bill,Clinton*), the frequency f_A of the first name at the first position in the bigram, regardless of the word at the second position, i.e. (*Bill,**), and the frequency f_B of the family name at the second position in the bigram, regardless of the word at the first position, i.e (**,Clinton*). Conditional probabilities P(A|B), for the first name occurrences depending on the family name occurrences, and P(B|A), for the family name occurrences depending on the first name occurrences, are then calculated for each first-name-family-name pair.

4.1 Comparing Conditional Probabilities of Non-nominative Cases Mutually and with Conditional Probabilities of Nominative

First we mutually compared conditional probabilities P(A|B) of the three inflective (i.e non-nominative) cases between themselves, in order to see if the probability for one particular case is either dominantly high or low (Table 3, left side). For 44 of the 100 persons the genitive/accusative morpheme was the most informative (i.e. had the highest probability), whereas the instrumental morpheme had the highest informativeness in 45 cases. The dative/locative morpheme was by far the least dominant: it appeared to be the most informative morpheme in only 11 of the 100 cases. The same comparison was performed for the probability P(B|A), with almost identical results (Table 3, right side): the genitive/accusative and the instrumental morpheme are the most informative in slightly more than 40% of cases while the dative/locative morpheme keeps the low profile of being the most informative in less than 20% of total cases.

Table 3. Number of occurrences as the highest probability among inflective cases

P_{GA}(A|B)	P_{DL}(A|B)	P_I(A|B)	P_{GA}(B|A)	P_{DL}(B|A)	P_I(B|A)
44	11	45	41	16	43

Next, we compared the probabilities P(A|B) and P(B|A) of each of the three inflective cases with those of nominative (Tables 4 and 5).

Considering the probability P(A|B) we note that the genitive/accusative and instrumental morphemes are more informative than the nominative morpheme in more than 60% of occurrences. For the opposite direction, the probability P(B|A), this dominance is even larger: more than 70% of total occurrences. On the other hand, the dative/locative morpheme is more informative than the nominative morpheme in less than 50% of total occurrences (nominative is particularly dominant for the direction P(A|B)). We can make a general conclusion that genitive/dative and instrumental are more informative than nominative, and that this informativeness is higher when given a first name and looking for a family name (i.e. P(B|A)) than in the case when the family name is known and we look for the first name.

Table 4. Comparing the number of occurrences as the more informative morpheme between each non-nominative case and nominative considering probability P(A|B)

case	$P_{inflective}$(A|B)	P_N(A|B)
G/A	64	36
D/L	36	64
I	62	38

Table 5. Comparing the number of occurrences as the more informative morpheme between each non-nominative case and nominative considering probability P(B|A)

case	$P_{inflective}$(B|A)	P_N(B|A)
G/A	78	22
D/L	48	52
I	71	29

4.2 Using Normalized Pointwise Mutual Information as Collocation Measure

Pointwise mutual information (sometimes simply called *mutual information*, MI) [3] is a measure of information overlap between two random variables [1]. It can be interpreted as the number of bits of *shared information* between two words and *exemplifies two general conventions for association scores that all collocation measures should adhere to* [5]: (1) higher scores indicate stronger attraction between words or greater degree of collocativity, (2) the measure distinguishes between positive associations (where co-occurrence frequency O is greater than expected frequency E) and negative associations (where O < E), assigning positive and negative scores, respectively.

MI measures *how much the actual probability of a particular co-occurrence of events p(x,y) differs from what we would expect it to be on the basis of the probabilities of the individual events assuming their independence* [2]:

$$MI(x,y) = \ln \frac{p(x,y)}{p(x) \cdot p(y)} \tag{1}$$

The MI measure can be normalized to have its values within the [-1, 1] interval. Fixing the bounds gives us a more intuitive comparison between the inflective cases and the nominative case. The *normalized pointwise mutual information* (NPMI) [2] is defined as:

$$MI_N(x,y) = \frac{\ln \frac{p(x,y)}{p(x) \cdot p(y)}}{-\ln p(x,y)} \tag{2}$$

We calculated an average score of the NPMI measure for all 100 first-name-family-name test pairs in nominative and in the inflective cases (Table 6). As expected, the score was highly positive (i.e. implying a high degree of positive collocation) for all cases. However, the results clearly show that collocation is stronger for all three inflective morpheme pairs in comparison with the nominative morpheme pair. The highest score was obtained for the instrumental case, followed by genitive/accusative and dative/locative. The nominative case had the smallest collocation score.

Table 6. The average value of the NPMI measure for each of the four morpheme types

	Nom.	Gen./Acc.	Dat./Loc.	Inst.
Avg NPMI	*0.7358*	0.7954	0.7728	**0.8141**

Applying other measures may show results that substantially differ from those obtained using NPMI. Evert [5] states that the choice of the collocation measure always depends on the task. For instance, if we search for the strongest collocation pairs in the entire corpus, we should avoid measures that prefer assigning high scores to low-frequency pairs, such as MI (or NPMI). The English words *curriculum* and *vitae* almost never appear separated (thus having a high MI score), but are much less frequent altogether in a corpus than the paragon example of collocation *ring-bell* (the pair *ring-bell* has smaller MI, since *ring* and *bell* have a much stronger tendency to appear separately than *curriculum* and *vitae*). Measures designed to favor such "general importance" of the candidate pairs are called *significance measures*, whereas MI is an *effect measure* [5]. Significance measures are prone to high-frequency bias and ask the question: "how strongly are the words attracted to each other?" Effect measures ask the question: "how much evidence is there for a positive association between words, no matter how small effect size is?" The latter question is the one we ask in our experiment. We are interested in the association between the first and the family name in a particular case, not the significance (i.e. total occurrence frequency) of the name in that particular case.

As an illustration of how a wrong choice of the collocation measure could violate the results, we calculate the average value of the simple logarithmic likelihood (*simple-ll*) measure for each of the four morpheme types (Table 7). This significance measure is calculated as:

$$\text{simple-ll} = 2 \cdot \left(O \ln \frac{O}{E} - (O - E) \right) \quad (3)$$

The fact that the nominative morphemes are substantially more frequent then each of the inflective morphemes results in the highest simple-ll score for nominative.

Table 7. The average value of the simple-ll measure for each of the four morpheme types

	Nom.	**Gen./Acc.**	**Dat./Loc.**	**Inst.**
Avg simple-ll	**7826.17**	4144.34	746.38	924.73

4.3 Two-Sample Chi-Squared Test

The numbers shown in Tables 4, 5, and 6 tell that non-nominative personal name bigrams are more collocative and some of them more informative than the nominative bigrams. However, these results may not be statistically significant, i.e. they may result from a difference situated within the margin of error.

We can consider two cases (two morpheme types) as two different samples. In each of these "samples" one component of the personal name appears f_{AB} times with the other component and $f_A - f_{AB}$ (given the first name and measuring the occurrence probability of the family name) or $f_B - f_{AB}$ times without it (given the family name and measuring the occurrence probability of the first name). This results in two contingency tables (Table 8).

Table 8. Contingency tables for the Pearson's chi-squared test

$f_{AB\text{-NOM}}$	$f_{AB\text{-OTHER}}$
$f_{A\text{-NOM}} - f_{AB\text{-NOM}}$	$f_{A\text{-OTHER}} - f_{AB\text{-OTHER}}$

$f_{AB\text{-NOM}}$	$f_{AB\text{-OTHER}}$
$f_{B\text{-NOM}} - f_{AB\text{-NOM}}$	$f_{B\text{-OTHER}} - f_{AB\text{-OTHER}}$

We state the following null-hypothesis: the number of occurrences the first name (and the family name, respectively) is followed by the corresponding family name (and the corresponding first name, respectively) and the number of occurrences the same first name is followed (and the same family name, respectively) by some other word *do not differ* for nominative and the other case(s). Thus, the resulting different values of P(A|B), P(B|A) and NPMI for different cases would not be significant.

The Pearson's chi-squared test (with the probability threshold 0.001, $t_{0.001} = 10.83$) shows that the null hypothesis can be rejected for all three non-nominative morpheme types and for both directions considering the family name and first name (Table 9).

Table 9. Results of the Pearson's chi-squared rejecting the null hypothesis

	N vs. G/A	N vs. D/L	N vs. I
first name → last name ($f_A - f_{AB}$)	1399.80	96.08	764.76
last name → first name ($f_B - f_{AB}$)	1459.10	303.54	480.18

5 Conclusion

In this paper we have analyzed the inflective aspects of Croatian personal name bigrams consisting of a first name and a family name. We handpicked names of 100 male persons, each of them making at least 10 million hits on the Google ™ search engine and producing four different declension morphemes. Using the normalized pointwise mutual information to quantify the collocation between the first name and the family name, we proved that the two components of a personal name are more likely to co-occur in non-nominative cases than in nominative. On the other hand, the conditional probabilities of finding a proper name bigram, given that one of its parts is a first name or a family name are larger for genitive/accusative and instrumental than for nominative, but smaller for dative/locative than nominative. The results of the Pearson's chi square test reject the possibility that higher collocation for non-nominative cases is statistically not significant.

In our future work, we will extend our study to bigrams consisting of a noun and its modifying prenominal adjective. We will also implement a new software module for Hascheck, which will exploit non-nominative declension forms to learn proper names.

References

1. Baroni, M., Evert, S.: Statistical Methods for Corpus Exploitation. In: [11], article 36
2. Bouma, G.: Normalized (Pointwise) Mutual Information in Collocation Extraction. In: Proc. GSCL Conf. 2009, pp. 31–40 (2009)
3. Church, K., Hanks, P.: Word Association Norms, Mutual Information and Lexicography. Computational Linguistics 16(1), 22–29 (1990)
4. Dembitz, Š., Randić, M., Gledec, G.: Advantages of Online Spellchecking: a Croatian Example. Software – Practice & Experience 41, 1203–1231 (2011)
5. Evert, S.: Corpora and Collocations. In: [11], article 58

6. Fellbaum, C. (ed.): WordNet. An Electronic Lexical Database. MIT Press, Cambridge (1998)
7. Hascheck. Hrvatski Akademski Spelling Checker, `http://hascheck.tel.fer.hr` (retrieved December 11, 2011)
8. Jurafski, D., Martin, J.H.: Speech and Language Processing. Prentice Hall, Englewood Cliffs (2000)
9. Krstev, C., Vitas, D., Gucul, S.: Recognition of Personal Names in Serbian Texts. In: Proc. RANLP 2005, pp. 288–292 (2005)
10. Levenshtein, V.I.: Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. Soviet Physics Doklady 10, 707–710 (1966)
11. Lüdeling, A., Kytö, M. (eds.): Corpus Linguistics. An International Handbook. Mouton de Gruyter, Berlin (2008)
12. Nadeau, D., Sekine, S.: A Survey of Named Entity Recognition and Classification. Linguisticae Investigationes 30(1), 3–26 (2007)
13. Piskorski, J.: Named-Entity Recognition for Polish with SProUT. In: Bolc, L., Michalewicz, Z., Nishida, T. (eds.) IMTCI 2004. LNCS (LNAI), vol. 3490, pp. 122–133. Springer, Heidelberg (2005)
14. Popov, B., Kirilov, A., Maynard, D., Manov, D.: Creation of Reusable Components and Language Resources for Named Entity Recognition in Russian. In: Proc. LREC 2004, pp. 309–312 (2004)
15. Sag, I.A., Baldwin, T., Bond, F., Copestake, A.A., Flickinger, D.: Multiword Expressions: A Pain in the Neck for NLP. In: Gelbukh, A. (ed.) CICLing 2002. LNCS, vol. 2276, pp. 1–15. Springer, Heidelberg (2002)
16. Tsvetkov, Y., Wintner, S.: Identification of Multi-word Expressions by Combining Multiple Linguistic Information Sources. In: Proc. EMNLP 2011, pp. 836–845 (2011)

Multi-agent Negotiation of Virtual Machine Migration Using the Lightweight Coordination Calculus

Paul Anderson, Shahriar Bijani, and Alexandros Vichos

School of Informatics, University of Edinburgh, UK
{dcspaul,s.bijani}@ed.ac.uk, alexandros.vichos@gmail.com

Abstract. LCC is a Lightweight Coordination Calculus which can be used to provide an executable, declarative specification of an agent interaction model. In this paper, we describe an LCC-based system for specifying the migration behaviour of virtual machines in a datacentre. We present some example models, showing how they can be used to implement different policies for the machine allocation and migration. We then describe a practical implementation of the system which can directly execute the LCC specifications.

Keywords: autonomic computing, multi-agent systems, virtual machines, OpenKnowledge, Lightweight Coordination Calculus.

1 Introduction

Virtualisation technology has recently transformed the availability and management of compute resources. Each *physical machine* (PM) in a datacentre is capable of hosting several *virtual machines* (VMs). From the user's point of view, a virtual machine is functionally equivalent to a dedicated physical machine; however, new VMs can be provisioned and decommissioned rapidly without changes to the hardware. VMs can also be *migrated* between physical machines without noticeable interruption to the running applications. This allows dynamic load balancing of the datacentre, and high availability through the migration of VMs off failed machines. The resulting *virtual infrastructure* provides the basis for *cloud computing.*

Managing the placement and migration of VMs in a datacentre is a significant challenge; existing commercial tools are typically based on a central management service which collates performance information from all of the VMs. If the current allocation is unsatisfactory (according to some policies), then the management service will compute a new VM allocation and direct agents on the physical machines to perform the necessary migrations.

As the size and complexity of datacentres increases, this centralised management model appears less attractive; even with a high-availability management service, there is possibility of failure and loading problems. If we would like to extend the domain of the virtual infrastructure to encompass multiple datacentres,

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 124–133, 2012.

managed by different providers, then the central model is no longer appropriate; in this federated "cloud" scenario, there may no longer be a single organisation with ultimate authority over all of the infrastructure.

This motivates us to propose a less centralised solution where agents located on the physical machines negotiate to transfer VMs between themselves, without reference to any centralised authority. This seems particularly appropriate for many situations where a globally optimal solution is not necessary or feasible; for example, if a machine is overloaded, it is often sufficient to find some other machine which will take some of the load. Likewise, an underloaded machine simply needs to take on additional VMs to improve its utilisation; there is no need for any global knowledge or central control.

In this paper, we present an experimental implementation of the above scenario in which agents follow *interaction models* (IMs) described in the *lightweight coordination calculus* (LCC). The agents use the OpenKnowledge framework to locate appropriate interaction models and to identify suitable peers. These interaction models specify the agent behaviour, and allow them to make autonomous decisions; for example, the choice of VM to accept could be based on local capabilities, the properties of the VM being offered, the financial relationship with the donor, etc.

One important consequence of this approach is that we can very easily change the global policy of an entire infrastructure by introducing new interaction models. For example, a particular model may encourage the physical machines to distribute the load evenly among themselves; this makes a lightly-loaded infrastructure very agile and able to accept new VMs very quickly. Alternately, a different interaction model may encourage the machines to prefer a full, or empty, loading as opposed to a partial one. Some of the machines would then be able to dispose of all their VMs, allowing them to be turned off and hence saving power.

2 LCC and OpenKnowledge

A computational agent - such as one responsible for one of our physical machines - must be capable of acting autonomously, but it will also need to communicate with other agents in order to achieve its goals. In a multi-agent system (MAS), the agents often observe conventions which allow them to co-operate. These are analogous to the *social norms* in human interactions, and may be more or less formal – an oft-cited example is the rules which govern the bidding process in an auction. In our application, agents must be able to compare the respective resource utilisation of their hosts, and reach an agreement about the transfer of a virtual machine. Typically, the social norms in a MAS will be defined using an explicit protocol. The *lightweight coordination calculus* (LCC) is a declarative, executable specification language for such a protocol.

LCC is based on a process algebra which supports formal verification of the interaction models. In contrast with traditional specifications for *electronic institutions*, there is no requirement to predefine a "global" script which all agents follow - the protocols can be exchanged and evolved dynamically during the conversation. LCC is used to specify "if" and "when" agents communicate; it does not define how the communication takes place[1], and it does not define how the agents rationalise internally. There are several different implementations of the LCC specification, including OpenKnowledge (see below), Li[2], UnrealLCC[3] and Okeilidh[4].

There is insufficient space here to describe the LCC language in detail; the OpenKnowledge website contains a good introduction[5], and there are also some video tutorials[6]. The following brief summary should be sufficient to follow the annotated examples presented in the next section:

Each IM includes one or more clauses, each of which defines a *role*. Each role definition specifies all of the information needed to perform that role. The definition of a role starts with: `a(`*roleName, PeerID)*. The principal operators are outgoing message (`=>`), incoming message (`<=`), conditional (`<-`), sequence (`then`) and committed choice (`or`). Constants start with lower case characters and variables (which are local to a clause) start with upper case characters. LCC terms are similar to Prolog terms, including support for list expressions. Matching of input/output messages is achieved by structure matching, as in Prolog.

The right-hand side of a conditional statement is a *constraint*. Constraints provide the interface between the IM and the internal state of the agent. These would typically be implemented as a Java *component* which may be private to the peer, or a shared component registered with a discovery service.

OpenKnowledge (OK[7])[7,12] provides an implementation of LCC, together with some additional functionality, including a distributed *discovery service*. Peers register their desired roles with the service, and this identifies a suitable set of peers to engage in a particular interaction. The peers are then notified and the interaction proceeds without further involvement of the discovery service[8]. OK is capable of quite sophisticated ontology-based role matching, and negotiation among the peers when attempting to fill the roles for a particular interaction. However, we only make use of the basic matching functions - for example, matching an "overloaded" peer with an "underloaded" one.

The OK discovery service also provides facilities for discovering and distributing both interaction models and components (OKCs). This means that a physical machine (in our application) need only register its willingness to participate, and

---

1 The inter-agent communication mechanism is defined by the implementation.
2 `http://sourceforge.net/projects/lij`
3 `http://sourceforge.net/projects/unreallcc`
4 `http://groups.inf.ed.ac.uk/OK/drupal/okeilidh`
5 `http://groups.inf.ed.ac.uk/OK/index.php?page=tutorial.txt`
6 `http://stadium.open.ac.uk/stadia/preview.php?whichevent=984&s=29`
7 `http://groups.inf.ed.ac.uk/OK/`
8 In practice, one of the peers is elected as a coordinator for the interaction, and the coordinator executes the IM, only making calls to other peers when it is necessary to evaluate a constraint.

the behaviour will then be defined by the IMs and OKCs which are retrieved from the discovery service. Of course, individual machines remain free to fulfil their role in whatever way is appropriate - possibly by using their own IMs and/or OKCs rather than those available on the central discovery service. This allows particular machines, or federated groups to engage in the same interactions, using different policies.

3 Interaction Models for VM Migration

In this section, we describe two LCC interaction models, which implement two different VM migration policies:

– in the first policy, VMs migrate from busy peers to underloaded peers to balance the load of each peer (Figure 1).
– in the second policy, VMs migrate from underloaded peers to busy peers to fully use the resources of some peers and release the others.

Figure 1 illustrates the state diagram of the first policy. There are three states: *idle, overloaded* and *underloaded*. The *idle* state is the initial and the goal state, in which the peer is balanced. Each peer is assumed to be balanced at the beginning of the interaction. It may then change state, based on its status. Each overloaded peer interacts with an underloaded peer (if any exist), in order to balance the load.

```
// Definition of the “idle” role. Here, “idle” means the “balanced” state
a(idle, PeerID) ::
   // the constraint to check the state of the peer
   null <- getPeerState(Status) then
   // select the next state based on the peer’s status
   ( null <- isOverLoaded() then // if the peer is overloaded,
   // change the peer’s role to “overloaded” and pass the status
   a(overloaded(Status), PeerID)
   )  or
   ( null <- isUnderLoaded() then // if the peer is underloaded,
   a(underloaded(Status), PeerID) // change the role to “underloaded”
   )  or
   a(idle, PeerID) // otherwise, remain in the idle role (recursion)

// Definition of the “overloaded” role. “Need” is the amount of resources required
a(overloaded(Need), ID1) ::
   // send the “readyToMigrate(Need)” message to an underloaded peer
   readyToMigrate(Need) => a(underloaded, ID2) then
  // wait to receive “migration(OK)” from the underloaded peer
   migration(OK)<= a(underloaded, ID2) then
   // do the migration: send VMs from this peer to the underloaded peer
   null <- migration(ID1, ID2) then
   a(idle, ID1) // change the peer’s role to “idle”
```

```
// Definition of the "underloaded" role: "Capacity" is the amount of free resources
a(underloaded(Capacity), ID2) ::
   // receive the "readyToMigrate(Need)" message from an overloaded peer
   readyToMigrate(Need)<= a(overloaded, ID1) then
   // send back the "migration(OK)" message, if the migration is possible
   migration(OK) => a(overloaded, ID1)
                      <- isMigrationPossible(Capacity, Need) then
   null <- waitForMigration() then
   a(idle, ID2) // change the peer's role to "idle"
```

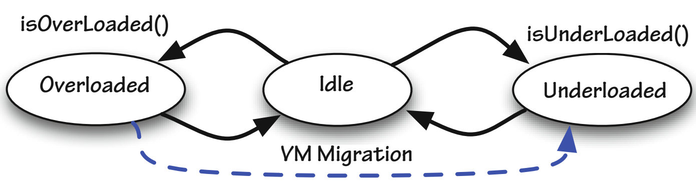


Fig. 1. The state diagram of the first migration policy: unbalanced peers interact to balance their loads

It may be that we would prefer to have the minimum number of active peers, each using almost all of their resources (e.g. to minimise the cost). This change of policy could be easily deployed by changing only some parts of the IM in the above LCC code. The following IM is an example of the second policy - it is very similar to the first policy, but it has one more state (*shutdown*) for the free peers with no load:

```
// In this policy, "idle" means the fully-loaded state, which is the goal state.
//     i.e the peer uses all available resources
a(idle, PeerID) ::
   null <- getPeerState(Status) then
   // if the peer's load > threshold (e.g. %50 ), but it still has free resources
   ( null <- canAcceptMoreLoad() then
     // change the peer's role to "notFullyLoaded" and send the peer's status
     a(notFullyLoaded(Status), PeerID)
   ) or
   ( null <- isUnderLoaded() then // if the peer is underloaded (e.g.≤ %50)
     a(underloaded(Status), PeerID) // change the role to "underloaded"
   ) or
   ( null <- hasNoLoad() then // if the peer has no load
     a(shutdown, PeerID) // change the role to "shutdown"
   ) or
   a(idle, PeerID) // otherwise, the peer is fully-loaded (recursion)
```

```
// Definition of the "notFullyLoaded" role. "Capacity" = free resources
a(notFullyLoaded(Capacity), ID1) ::
   // send the "readyToMigrate(Capacity)" message to an underloaded peer
   readyToMigrate(Capacity) => a(underloaded, ID2) then
  // wait to receive "migration(OK)" from the underloaded peer
   migration(OK)<= a(underloaded, ID2) then
  // VM migration from an underloaded peer to this peer (notFullyLoaded)
   null <- migration(ID2, ID1) then
   a(idle, ID1) // change the peer's role to "idle"

// Definition of the "underloaded" role. "Load" is the amount of busy resources
a(underloaded(Load), ID2) ::
  // receive the "readyToMigrate(Capacity)" message from a notFullyLoaded peer
   readyToMigrate(Capacity)<= a(notFullyLoaded, ID1) then
   // send back the "migration(OK)" message, if the migration is possible
   migration(OK) => a(overloaded, ID1)
                       <- isMigrationPossible(Capacity, Load) then
   null <- waitForMigration() then
   a(shutdown, ID2) // change the peer's role to "shutdown"

// Definition of the "shutdown" role
a(shutdown,ID3)::
   null <- releaseResources() then // the peer's resources will be released
   null <- sleep(5000) then // wait 5 second
   a(idle, ID3) // after wake up, change the peer's role to "idle"
```

The level granularity of the LCC code in reflecting the details of the policy implementation is optional. Instead of implementing the details of the policy (e.g. delays, etc.) in LCC code we could push them down into the Java constraints.

4 A Prototype

To validate the approach in a realistic environment, we constructed a small prototype, based on a cluster of physical machines (see figure 2). This consisted of four HP Proliant DL120 G6 servers each with 4GB memory, and each capable of supporting in the order of 10 virtual machines. The configuration of this cluster is described fully in [13]. Briefly:

1. We chose to use KVM[9] as the default hypervisor. This is a loadable kernel module that converts the Linux kernel into a bare metal hypervisor. It is a mainstream component of Linux distributions (Fedora and Redhat), it is freely available, well-supported locally, and has a programmable interface via `libvirt`. We would expect the prototype to be equally implementable on top of other common hypervisors such as VMware or Xen.

[9] `http://www.linux-kvm.org/`

2. We used the `libvirt`[10] library to provide a programmable API to the underlying hypervisor (KVM). This supports full remote management of the virtual machines, including stop/start and migration. The KVM C API was exposed to our Java components using JNA.
3. Shared access to the VM images is necessary for migration, and we used an NFS filesystem to store the images.
4. We wrote code to interface OpenKnowledge components (and/or peers) with both `libvirt` and the underlying OS facilities. This provided control of the virtual machines, and information on the state of the system - such as memory and CPU usage.
5. We used the standard OpenKnowledge discovery service to provide peer discovery and distribution of the components and interaction models. A single discovery server was adequate in this case, although OpenKnowledge uses the Pastry overlay network which is capable of supporting a redundant peer-to-peer network of connected discovery servers.

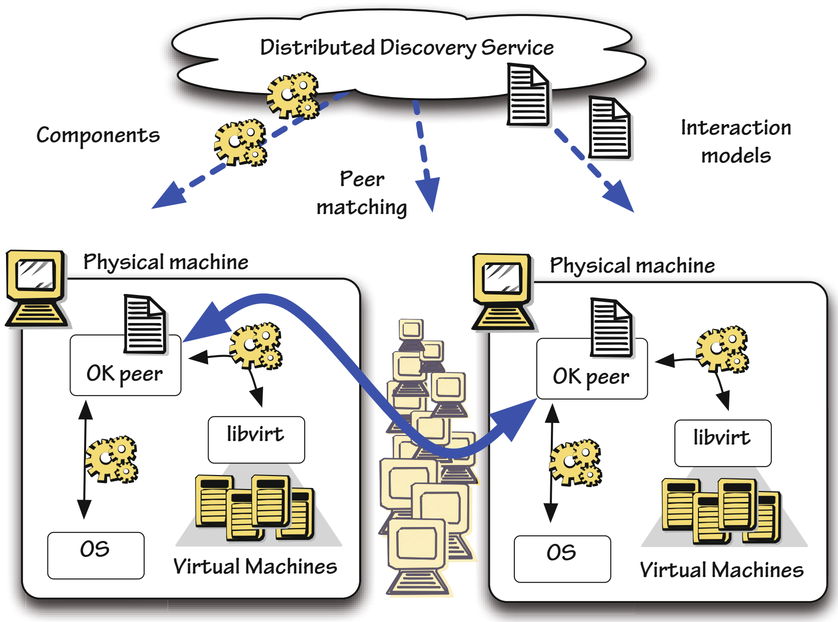


Fig. 2. The architecture of the prototype described in section 4

5 Evaluation

In addition to the experiments with the live prototype, we used a simple simulator to investigate the behaviour of more complex models with a larger number of

[10] http://libvirt.org/

machines and more controlled loading. Figure 3 shows the results of 50 simulated virtual machines running on 15 physical machines. In this example, physical machines offload VMs if they have a load greater than 120% of the average, and they accept VMs if they have a load less than 80%. Initially, the VMs are allocated randomly and the resulting load is uneven.The system stabilises after a time with all the physical machines except one within the desired range (the load on the remaining machine cannot be reduced because all of the machines have a load greater than 80%). Further results and details of the simulator are available in [5].

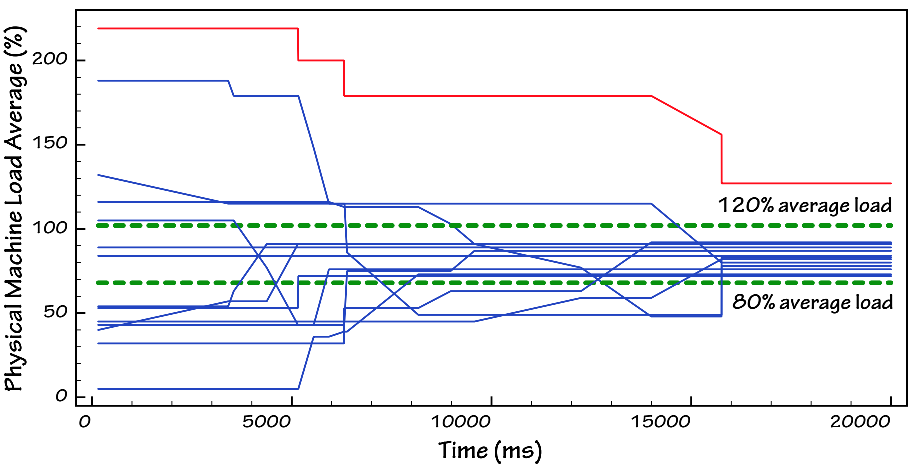


Fig. 3. A simulation showing the load on 15 physical machines as they interact to balance a load of 50 virtual machines

6 Related Work

There is a considerable amount of existing work on load balancing of virtual infrastructures (see, for example [1,2,15]). Most of this work assumes a central service which collects monitoring data from the physical and virtual machines, computes any necessary re-allocation, and orchestrates the appropriate migrations. However, this leads to difficulties in managing the interactions of imperative control algorithms[10], and limits the degree to which it is possible to exploit the resources of a more federated environment[3,9].

VMWare is a popular provider of commercial management infrastructure for virtual datacentres. The VMWare *vSphere Distributed Resource Scheduler* (DRS) product allows the user to specify rules and policies to prioritise how resources are allocated to virtual machines. DRS[11] "continuously monitors utilisation across resource pools and intelligently aligns resources with business needs" . *vSphere*

[11] http://www.vmware.com/pdf/vmware_drs_wp.pdf

Distributed Power Management (DPM) allows workloads to be consolidated onto fewer servers so that the rest can be powered-down to reduce power consumption. Citrix Essentials[12] and Virtual Iron "Live capacity" [13] are other commercial products offering similar functionality, and LBVM[14] is an open-source product based on Red Hat Cluster Suite. However, all of these products use a centralised management model.

As noted by Kephart and Walsh[4], agent-based technologies are a natural fit for implementing *autonomic* systems[6]; CatNets[11], for example, is a market-based resource management system. Several people have applied agent-based techniques to virtual machine management: Xing[16] describes a system where "each virtual machine can make its own decision when and where to migrate itself between the physical nodes" - for example, two VMs may notice that the applications running on them are communicating frequently, and the VMs may decide that they should attempt to migrate so that they are physically closer. Spata and Rinaudo[8] describe a FIPA-compliant system with very similar objectives to our own which is intended to load-balance VMs across a cluster. However, we are not aware of any other implementation which is driven directly from a declarative specification of the interaction model.

7 Conclusions and Future Work

We have demonstrated that an agent-based approach using LCC interaction models is a viable technique for negotiating virtual machine placement and migration. This is especially appropriate where the number of machines involved is very large and global knowledge is neither possible, nor necessary. It is also applicable in federated situations where there is no single point of control, or policy.

We have only described comparatively simple interactions, but the abstraction provided by the LCC model makes this an ideal basis to explore more complicated scenarios. These might involve more sophisticated negotiations (such as auctions), and/or more dimensions to the underlying metrics (such as memory usage, bandwidth, proximity, etc.).

Acknowledgements. We would like to thank HP Labs Bristol, UK for providing the hardware for the experimental cluster, and Jun Le for allowing us to use the data from his simulations.

References

1. Bobroff, N., Kochut, A., Beaty, K.: Dynamic placement of virtual machines for managing SLA violations. In: 10th IFIP/IEEE International Symposium on Integrated Network Management, IM 2007, pp. 119–128 (Yearly 21-25, 2007)

[12] https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPE4XSE

[13] http://www.storageengineers.com/pdf_virtualiron/Evaluation_Guide_0107.pdf

[14] http://lbvm.sourceforge.net/

2. Bodík, P., Griffith, R., Sutton, C., Fox, A., Jordan, M., Patterson, D.: Statistical machine learning makes automatic control practical for internet datacenters. In: Proceedings of Workshop on Hot Topics in Cloud Computing (HotCloud) (2009)
3. Grit, L., Irwin, D., Yumerefendi, A., Chase, J.: Virtual machine hosting for networked clusters: Building the foundations for "autonomic" orchestration. In: Virtualization Technology in Distributed Computing, VTDC 2006, p. 7 (November 2006)
4. Kephart, J., Walsh, W.: An artificial intelligence perspective on autonomic computing policies. In: Proceedings of Fifth IEEE International Workshop on Policies for Distributed Systems and Networks, POLICY 2004, pp. 3–12 (June 2004)
5. Li, J.: Agent-based management of Virtual Machines for Cloud infrastructure. Master's thesis, School of Informatics, University of Edinburgh (2011)
6. Murch, R.: Autonomic Computing, 1st edn. IBM Press (2004)
7. Pinninck, A.P.D., Kotoulas, S., Siebes, R.: The OpenKnowledge kernel. In: Proceedings of the IX CESSE Conference (2007)
8. Spata, M.O., Rinaudo, S.: Virtual machine migration through an intelligent mobile agents system for a cloud grid. Journal of Convergence Information Technology 6 (June 2011)
9. Ruth, P., Rhee, J., Xu, D., Kennell, R., Goasguen, S.: Autonomic live adaptation of virtual computational environments in a multi-domain infrastructure. In: IEEE International Conference on Autonomic Computing, ICAC 2006, pp. 5–14 (June 2006)
10. Schmid, M., Marinescu, D., Kroeger, R.: A Framework for Autonomic Performance Management of Virtual Machine-Based Services. In: Proceedings of the 15th Annual Workshop of the HP Software University Association (June 2008)
11. Schnizler, B., Neumann, D., Veit, D., Reinicke, M., Streitberger, W., Eymann, T., Freitag, F., Chao, I., Chacin, P.: Catnets deliverable 1.1: Theoretical and computational basis. Tech. rep., CatNet Project (2005)
12. Siebes, R., Dupplaw, D., Kotoulas, S., Perreau de Pinninck, A., van Harmelen, F., Robertson, D.: The OpenKnowledge System: An Interaction-Centered Approach to Knowledge Sharing. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 381–390. Springer, Heidelberg (2007)
13. Vichos, A.: Agent-based management of Virtual Machines for Cloud infrastructure. Master's thesis, School of Informatics, University of Edinburgh (2011)
14. Walton, C., Robertson, D.: Flexible multi-agent protocols. Tech. rep., University of Edinburgh (2002)
15. Wood, T., Shenoy, P., Venkataramani, A., Yousif, M.: Black-box and gray-box strategies for virtual machine migration. In: Proceedings of the 4th Usenix Symposium on Networked Systems Design and Implementation. Usenix (April 2007)
16. Xing, L.: A Self-management Approach to Service Optimization and System Integrity through Multi-agent Systems. Master's thesis, University of Oslo, Department of Informatics (May 2008)

On-Line Communities Making Scense: A Hybrid Micro-Blogging Platform Community Analysis Framework

Cheng-Lin Yang and Yun-Heh Chen-Burger

Artificial Intelligence Applications Institute, Centre for Intelligent Systems and Their Applications, School of Informatics, University of Edinburgh
{s0969605,jessicac}@inf.ed.ac.uk

Abstract. The upsurge of Micro-blogging platform attracts enterprises to use it as a public relationship tool. It also act as a new form of news source, journalists can hunt for next upcoming breaking news. It is worth to identify communities from it and reveal social relationships among community members in a timely manner. However, traditional SNA approaches are insufficient to achieve the requirement in a reasonable time. In this paper, we proposed a hybrid framework to tackle the problem. It is designed to identify the community with real social relationships automatically, that withstand dynamically changing content, have the ability to process fast and live-streaming data and provide a self-feedback mechanism to refine the result without human interference. The benefit of this framework is that average users should be able to employ it and to really understand communities in micro-blogging platforms without any or limited domain knowledge.

Keywords: Social Network Analysis, Micro-blogging System, Machine Learning.

1 Introduction

What is community? Traditionally speaking, a community is a group of people who are gathered to embrace the same values or share the same responsibility. From a geographical perspective, a community can be defined by a street, a town or a specified area [1]. From the relationship aspect, a community is formed by human interaction [2]. To summarise, a community is a group of people who have the same values and live in a specified area. Members of the community feel dependence and through social interactions that build up the collective spirits.

With the rapid development of communication technology, the Internet has become an indispensable utility in daily life. People exchange information and knowledge on the Internet through various devices, forming a large social network and developing different types of on-line communities. The user with new communication technology uses the forum or blog system to share his knowledge or experience with multimedia resources. By using search engine such as Yahoo

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 134–143, 2012.

and Google, the user who is interested in particular topics can reach the information much quicker and easier. He is able to discuss the topic with users from different countries by Internet. Therefore, a new type of community is formed. In this paper, we call them on-line communities.

Members of an on-line community are not restricted to the same geographical area unlike the community as defined in the traditional sense. An on-line community can be defined as a social phenomenon formed by a group of people who communicate with each other through the Internet. The interaction on the Internet satisfies people's interests and fantasies as well as developing social relationships [7].

1.1 Upsurge of Micro-blogging Platform

Micro-blogging platforms such as Twitter [3], Yahoo meme [4], Plurk [5] and Sina Weibo [6] have gained popularity since 2007. They quickly attracted the attention of many users, including many celebrities, politicians, and television and sports stars, who to share their daily life and personal opinions. Since the growth rate of registered users is staggering, enterprises have begun using it as a public relationship tool to announce news or provide the latest promotion information. These platforms also act as a new form of news source, journalists are able to hunt for next upcoming breaking news. For example, the first report of the emergency landing of US Airway airplane on the Hudson River was posted on Twitter.

When using a traditional blog system, e.g., Blogger and Wordpress, the user has to organize several paragraphs to form a blog-post, which becomes a problem if the user wants to share his current feelings in only a few words or sentences or to share interesting photos with a few comments. Posts like this would be considered odd by blog viewers since they expect a full article rather than a few sentences or photos without descriptions. A Micro-blogging system successfully fills the need of this kind of user. It is called "micro" because it normally limits either the word count or number of characters to 200 words. In addition, unlike the traditional blog system, where the user has to compose the article on personal computer, a micro-blogging system makes use of various communication technologies. The user can submit a micro-blog post from his desktop, laptop, tablet, smartphone and even by sending SMS. An individual micro-blog post can also contain video or image links with a few comments. Friends of micro-blog will be notified automatically by the system, so they can respond to the post immediately after it is posted.

2 Motivation and Challenges

Most micro-blogging posts are public, which means that everyone can view it and participate if they are interested in the topic. Any user can share his topics and let others participate in the discussion. These interactions form an on-line community. If we translate this behavior to the real world, most would find it very strange because people do not typically allow a stranger to join their daily life. The interesting way in which an on-line community forms through a micro-blogging system causes our curiosity. We want to understand the micro-blogging community deeply. However, we need to know the answers to several questions as follows: a) How is a micro-blogging community identified? b) How can a micro-blogging community be aligned with a human community, and how can we define those behaviours which cannot be aligned and understood? c) What kind of behaviour can be identified in a micro-blogging system? The challenges inherent in these questions are: a) The types of communities are complex, so how do we identify micro-blogging communities automatically, efficiently and accurately? b) The relationships of members lack definition, which means we need to carefully identify and interpret relationships that are important to the community. c) Unlike human communities, which have boundaries such as geographical boundaries, physical activities and gatherings, etc., on-line micro-blogging communities do not. Therefore, defining its boundaries will be challenging.

3 Related Work

The key to understanding the micro-blogging system is to identify communities behaviours of on-line communities. Social network analysis (SNA) is commonly used. SNA considers the human community and their relationships as a graph. Any node represents an individual, and the edge which connects two nodes indicates the relationship between two individuals [8]. By applying traditional clustering algorithms of graph theory like Hierarchical Clustering [9] and k-means [10], we can find the clusters in the given graph.

3.1 Traditional SNA Aprroach

Hierarchical Clustering treats each node as an unique cluster. It finds the most similar nodes, which is determined by the distance between two nodes calculated by the predefined distance function, and merges them as a new cluster. The procedure continues until all nodes are merged into new clusters. The benefits of Hierarchical Clustering are: 1) The algorithm is simple and easy to implement. 2) It does not require the centre point among all nodes. As long as we have the distance between the nodes, the algorithm is able to find the clusters in the given data. However, the drawbacks of Hierarchical Clustering are: 1) We can not predict how many clusters will be generated. 2) The computational complexity is $O(Ed\log N)$ where N is number of nodes and E is number of edges, which requires large amount of computing resource. When the given data is too large, the algorithm cannot generate the results in a reasonable amount of time.

Comparing the need of understanding the micro-blogging system, Hierarchical Clustering is able to generate a maximum number of clusters since it will merge the nodes close to each other into a cluster, which is beneficial since we cannot predict how many types of communities will emerge in the given data. However, in terms of the data size of a real micro-blogging system. Hierarchical

Clustering algorithm cannot deliver results in an acceptable timeframe, which renders this algorithm infeasible for our scenario since we need to identify the on-line communities in a realtime micro-blogging system.

In contrast, in k-means algorithm, k is the number of clusters we except to observe. Using that information, the algorithm will choose k data points randomly and consider those points as the centre point of clusters. After that, it calculates the distance from each data point to all centre points. Centre points then move to their closest data points. The procedure stops when no centre points can be moved. The benefits of k-means algorithm are: 1) Fast converge 2) The computational complexity is $O(NVk)$ where N is number of nodes and V is the number of vetoers, which means that requires less computing resource. Unfortunately, the drawback is that the result is easily affected by the initial centre point easily and gives a local maximum rather than a global one.

3.2 Local, Global and Score-Based Approach

Researchers have discovered that with traditional methods, the definition of community is not well-defined. Hence, the new methods try to analyse the network based on its density. They believe that the relationships within a community will be much denser than those relationships outside the community. Based on this concept, new methods are proposed by other researchers.

Local Methods. Since researchers believe that a community in a social network should have denser relationships, we should be able to find a subgraph that is closely related in the given graph. The most well-known method is to find the clique. A clique is a subset of a graph, which consists three or more nodes. Each node in the clique has a edge connected to all other nodes.

It would be meaningless if the discovered *clique* is too small. On the other hand, finding the largest clique is a NP-hard problem [11]. Moreover, the definition of clique is restricted. A clique can collapse if some of the edges are lost, which is easy to observe in practical data. In order to solve this problem, several clique relaxations were proposed, such as k-cliques [12], and k-clubs [13].

The original definition of clique restricts the distance from a node to all other nodes to be one. k-clique eases the distance limitation by requiring that the distances between any two nodes be less than k. Since k-clique does not require that the shortest path between two nodes must pass through k-clique itself, the shortest path may go through a node which does not belong to k-clique. Situations like this cause two problems: 1) The distances between k-clique nodes may be longer than k. 2) The nodes in k-clique are possibly not connected. In our scenario, the community identified by k-cliques may have loose relationships or even no relationship at all.

To overcome k-clique's problem, k-clubs [13] was proposed. The definition of k-clubs is that any distance between node pairs in k-clubs group must be less than k, and the paths of node pairs must go through k-clubs' group.

Global Methods. Instead of finding cliques, global methods such as the G-N algorithm focus on calculating relationship density [14]. The procedure of the G-N algorithm is to calculate the betweenness of each edge in the given graph. The betweenness is used to measure the importance of a node between any other two nodes.

G-N algorithm will purge the edge with the largest betweenness each time until the graph is partitioned into several subgraphs. It is obvious that when applying the G-N algorithm, the betweenness calculations are performed repeatedly, which costs extensive computing time. The time complexity of the G-N algorithm is $O(N^E)$ where N is number of nodes and E is number of edges. The G-N algorithm is not suitable to large scale network.

[15] consider the social network as an electrical circuit, with the edge of the graph containing different levels of resistances. The main idea of this approach is that human relationships are fading out from layers of friends, which is exactly like the voltage growing smaller and smaller via flowing through the power circuit. The procedure of this approach is to select a node and assign one volt to it. Then assign 0 volt to randomly selected nodes from the network. The voltages of all nodes are calculated by applying Kirchoff's Laws. The value of each node should between 0 to 1. After that, a threshold is given and nodes have the value higher than the threshold belong to one cluster and the rest nodes belong to another. It can also be extended to find multiple clusters by given range of threshold.The computational complexity of this method is $O(N+E)$ where N is number of nodes and E is number of edges. Although the speed of it is attractive to us, two drawbacks are shown: 1) The quality of generated results is based on iteration time rather than the size of the graph, which will make it difficult for us to find the best value of iterations. 2) The number of generated communities needs to be defined before the procedure, which is not suitable for us since the number of communities in a social network is unknown.

Score-Based Methods. Algorithms suchs as PageRank [16], Hypertext Induced Topic Selection (HITS) [17] and Betweenness Centrality[18] are also used by researchers. The concept of them is to compute an authority score based on the relationships of nodes in the network. Betweenness Centrality was used by [18] to identify terrorist groups. Also, J. Qin and et. al., [19] applied the PageRank algorithm in their hunt for the global Salafi Jihad network. However, the dataset they used were relatively small, and their target result had clearly predefined attributes, which makes they unsuitable to our scenario.

3.3 Finding by Text

Q.-M Li and et.al, [20] and D. Shen and et al [21] believed that if two individuals have the same interests, they are possible to be *hidden* friends or in the same community. Besides on this assumption, they then collected the blog posts from the Internet and applied Latent Semantic Indexing (LSI) [22] on each post.

Using keyword filters, the interests of users will be generated. Finally the users with the same interests will be considered to be in the same community. This approach lacks of consideration for user relationships, as people may have the same interests but never interact with other users, which make it hard for us to define them as being in the same community.

Using techniques such as LSI helps researchers to observe blog's features. Sometimes, the discovered features are dangerous, such as if they reveal that the owner of the tends to harbour racial discrimination or hatred of democracy or a particular country. However, the accurate level of danger of a blog owner cannot determined by his posts alone. M. Chau and J, Xu [23] believe that if a blog owner who has several dangerous features within his posts also recommends blogs or websites with dangerous features or add lots of blog owners who also have dangerous features as friends, he is highly likely to be a member of a criminal group.

4 Proposed Hybrid Framework

A hybrid system with a three-layered framework: collection, classification and reasoning layers. The architecture of proposed system is shown in Figure 1.

4.1 Collection Layer

The collection layer contains components that fetch the data from the micro-blogging system, process the raw data into a pre-defined format and convert the data into numerical parameters. The work will be performed by three components: crawler, database and data condenser. The requirement of them will be presented below.

Crawler. The crawler is responsible for retrieving the user data from the concerned micro-blogging system. All fetched data will be stored in the storage for further usage. It is designed to be a lightweight but targeted daemon so it can be deployed on multiple machines easily to increase the throughput.

The Storage. In order to support fast lookup and flexible schema, the proposed system will take advantage of a distributed key-value database system such as Cassandra [24] or MongoDB [25], which allows us to change the table schema without altering the entire table. Scalability is another concern for any system that handles tremendous amount of data. A traditional relational database, if to be used, needs to be well-designed before development and it may take a great deal of effort to expand the system scale. In contrast, a distributed database system provides a simple procedure to add new node into the system.

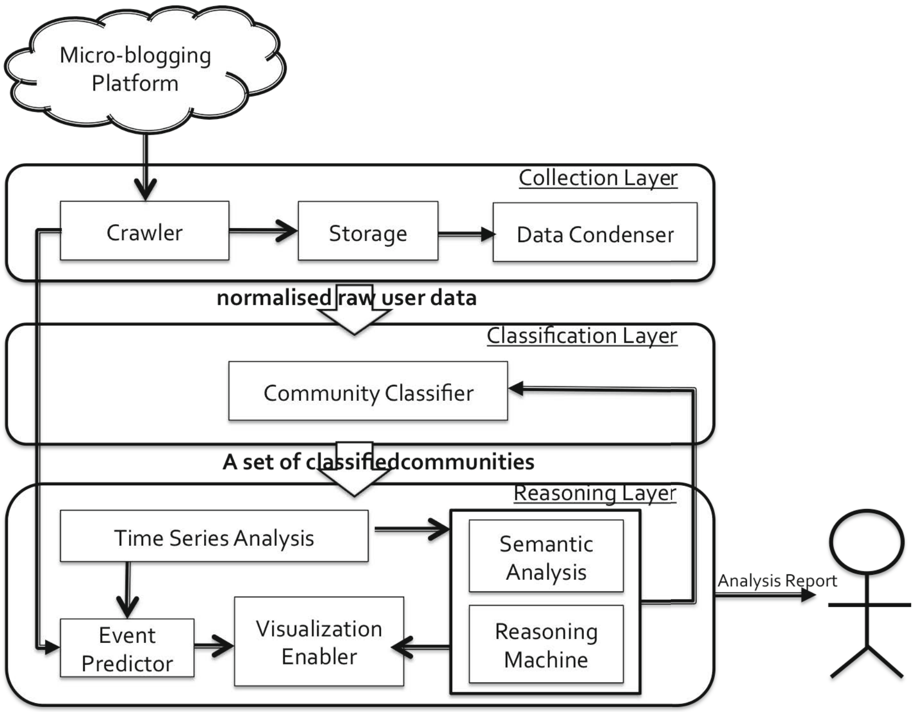


Fig. 1. Overview of proposed hybrid three-layered framework

Data Condenser. The data condenser reads the raw data from the database. The raw data contains noises like auxiliary words, emoticons or random characters, so, it is the data condenser's responsibility to remove these noises. It is also responsible for converting and normalising the selected fields such as relationships among users into numerical parameters for the classification layer.

4.2 Classification Layer

The classification layer is the main interface which processes numerical parameters from the collection layer. It generates a set of classified communities to the reasoning layer that allows users to manipulate the final results easily.

Community Classifier. The community classifier plays an important role in the proposed framework. The process flow can be summarised as follows. The crawler retrieves the raw data from the micro-blogging system and stores it in a database. After that, the data condenser fetches the raw data from the database. It removes all noises from the raw data and transforms the purified data into numerical parameters. Based on parameters, the community classifier applies the machine learning classification method to generate communities. This is then fed to the reasoning layer for further processing. The reasoning layer will analyse the results and text content with a logic-based reasoning engine, which will feedback to the community classifier to improve the quality of the result. Finally, the community classifier provides a finely tuned result to the user.

4.3 Reasoning Layer

The reasoning layer consists of a set of components which will infer the outcome of the classification layer based on its reasoning machine and feed the suggestion back to the upper layer. A framework is inaccessible if the user cannot visualise and manipulate the result. Hence, the reasoning layer also needs to provide a simple interface to the user.

Time Series Analysis. The timestamp of each user's responses to a conversation or statement is a crucial factor to a community. It reveals how the user interacts with the community as well as revealing the characteristics of each member. Therefore, we plan to carry out several time-based anaysis on user interactions, e.g. the result from the classification layer will be analysed on the timestamp field. This then will be fed to other components as an additional parameter.

Event Predictor. The value of an on-line community is that it reflects real daily life. The event predictor consists of a topic model based on observations from various micro-blogging conversations since each message within the conversation contains an unique timestamp. The topic model defines types of topics and how they grow through the timeline. By utilising the topic model, the event predictor will notify the user of which topics will be popular or even controversial in a given community.

Semantic Analysis. The focus of our research is to understand the on-line community. Therefore, the identified community requires further investigation. Traditional techniques cannot handle large amount of content within a reasonable time. We simplify the content into categories. By utilising the user generated hashtag and keyword spotting along with the pre-defined dictionary, we are able to achieve the simple categorization. It improves the performance and provides us a flexible way to adjust the category if needed.

Reasoning Machine. Reasoning machine is a logic rule-based inference engine with a set of rules which are given in advance and able to expand manually. It is used to verify the quality of the community generated by classification layer. The engine examines the social relationships among community members, how the user interacts with others and so on. The reasoning machine will evaluate all factors to see if it needs to ask the community classifier to change the parameter settings. It is also an interrupter which converts the query from visualize enabler into logic rules and returns the result to the user.

Visualize Enabler. A system will be inaccessible if the provided information cannot be easily understood by the user. The user may also want to make his own query to manipulate the result. The Visualize enabler tackles these problems by providing a simple user interface that makes the system as intuitive as possible.

5 Evaluation

Our proposed framework needs to be evaluated to see if it fits the requirements. The following criteria is designed to cover tests cases from different aspects.

1. Ground Truth
   A set of experts are invited to examine the generated result from our framework. The will examine the results and calculate the precision/recall rates to prove the accuracy of automatically classified communities generated by our framework.
2. Performance
   The test of performance is divided into two parts to cover overall execution and internal feedback speed. Candidate test cases to be considered are:
   – Feeding the realtime streaming data from the micro-blogging platform, it helps us to assess the overall performance on speed of our framework from the input to the final result. A threshold time is assigned and our system should generate the result faster than threshold time.
   – The design of our framework allows it adjust its model by reasoning machine. It should be agile enough to tune the model when new rules are applied. In this test, we will change the logic-based rule on reasoning machine. Our system should regenerate a new result in a reasonable time.
3. Robustness
   In a live micro-blogging system, its content usually contains certain noises such as spams or emoticons. Therefore, it is important to evaluate how accurate our framework to be performed under highly noise data. Our system needs to be able to handle the noise and the generated result should be acknowledged by human experts.
4. Usability
   A user interaction system should be manipulated by the user easily. Hence, in usability test, we will invite users who have rich and basic computer science background. Each user will score our framework from 1 to 5, which 1 means that he totally do not know how to use it and 5 represents that he feel trouble at all, after he uses the system. Our system should obtain 3.5 on average score on both groups.

References

1. Rothaermel, F.T., Sugiyama: Virtual internet communities and com- mercial success: individual and community-level theory grounded in the atypical case of timezone.com. Journal of Management 27(3), 297–312 (2001)
2. Koh, J., Kim, Y.G.: Knowledge sharing in virtual communities: an e-business perspective. Expert Systems 26(2), 155–166 (2004)
3. Twitter, http://www.twitter.com/
4. Yahoo meme, http://meme.yahoo.com/
5. Plurk, http://www.plurk.com/
6. Sina weibo, http://www.weibo.com/
7. Pliskin, N., Romm, C., Clarke, R.: communities and society: Toward and integrated three phase model. International Journal of Information Management 17(4), 261–270 (1997)
8. Wasserman, S., Faust, K.: Social network analysis: Methods and applications. Cambridge University Press (1994)
9. Everitt, B.S.: Cluster analysis (1979)
10. Hartigan, J.A.: Clustering algorithms. Probability and Mathematical Statistics (1975)
11. Garey, M., Johnson, D.: Computers and intractability: A guide to the theory of np-completeness. Freeman (1979)
12. Alba, R.: A graph-theoretic definition of a sociometric clique. Journal of Mathematical Sociology 3, 113–126 (1973)
13. Mokken, R.: cliques, clubs, and clans. Quality and Quantity 13, 161–173 (1979)
14. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences 99(12), 7821–7826 (2002)
15. Wu, F., Huberman, B.A.: Finding communities in linear time: A physics approach. The European Physical Journal B-Condensed Mater 38(2), 331–338 (2004)
16. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Stanford Digital Libraries Working Paper (1998)
17. Kleinberg, J.: Authoritative sources in a hyperlinked environment. In: Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 668–677 (1998)
18. Krebs, V.: Mapping networks of terrorist cells. Connections 24(3), 43–52 (2001)
19. Qin, J., Xu, J.J., Hu, D., Sageman, M., Chen, H.: Analyzing Terrorist Networks: A Case Study of the Global Salafi Jihad Network. In: Kantor, P., Muresan, G., Roberts, F., Zeng, D.D., Wang, F.-Y., Chen, H., Merkle, R.C. (eds.) ISI 2005. LNCS, vol. 3495, pp. 287–304. Springer, Heidelberg (2005)
20. Li, Q.-M., Xu, M.-W., Hou, J., Liu, F.-Y.: Web classification based on latent semantic indexing. Journal of Communication and Computer 3(1), 24–27 (2006)
21. Shen, D., Sun, J.-T., Yang, Q., Chen, Z.: Latent friend mining from blog data. In: Proceedings of the Sixth International Conference on Data Mining, pp. 552–561 (2006)
22. Deerwester, Dumais, Furnas, Lanouauer, Harshman.: Indexing by latent semantic analysis. Journal of the American Society for Information Science 41, 391–407 (1990)
23. Chau, M., Xu, J.: Mining communities and their relationships in blog: A study of online hate groups. International Journal of Human-Computer Studies 65(1), 57–70 (2007)
24. The apache cassandra project, http://cassandra.apache.org/
25. MongoDB, http://www.mongodb.org/

Measuring the Effectiveness of Throttled Data Transfers on Data-Intensive Workflows

Ricardo J. Rodríguez[1], Rafael Tolosana-Calasanz[1], and Omer F. Rana[2]

[1] Dpto. de Informática e Ingeniería de Sistemas
Universidad de Zaragoza, María de Luna 1, 50018 - Zaragoza, Spain
{rjrodriguez,rafaelt}@unizar.es
[2] School of Computer Science & Informatics
Cardiff University - Cardiff, United Kingdom
o.f.rana@cs.cardiff.ac.uk

Abstract. In data intensive workflows, which often involve files, transfer between tasks is typically accomplished as fast as the network links allow, and once transferred, the files are buffered/stored at their destination. Where a task requires multiple files to execute (from different previous tasks), it must remain idle until all files are available. Hence, network bandwidth and buffer/storage within a workflow are often not used effectively. In this paper, we are quantitatively measuring the impact that applying an intelligent data movement policy can have on buffer/storage in comparison with existing approaches. Our main objective is to propose a metric that considers a workflow structure expressed as a Directed Acyclic Graph (DAG), and performance information collected from historical past executions of the considered workflow. This metric is intended for use at the design-stage, to compare various DAG structures and evaluate their *potential* for optimisation (of network bandwidth and buffer use).

1 Introduction

Scientists can make use of workflow techniques, using often a combination of control/data flow graphs, to specify their experiments. There has been an increased interest in data-intensive scientific workflows recently, where the data between tasks in the workflow needs to be migrated across a distributed environment. Often the staging of data between tasks in such workflows is either assumed or the data transfer time is considered to be negligible (compared to task execution). Where data consists of files, previous approaches exploit data location and link bandwidth information to minimise data movement, but the transfer typically involves moving the output data of a task to its successor node as fast as possible after task completion [1]. However, when a task needs multiple files to be made available before it can begin execution, the task remains idle until all the required data files from other predecessor nodes are delivered. It is therefore not how fast each file can be moved to the task, but the interval from the delivery of the first file to the last one that is most significant. Even

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 144–153, 2012.

if the first file is delivered quickly, the task must still remain idle until others are also available, thereby leading to an ineffective use of *network bandwidth* and *buffer/storage* at the receiving task. If one file arrives too early taking up all of the network bandwidth for one task, it may be at a determent to other tasks (which may have to wait for their data to be delivered) – even though the receiving task still has to wait for other files. Similarly, if there is limited buffer capacity at the receiving task and the buffer needs to be shared between tasks, a quick delivery of one file (while waiting for other files to be delivered) for a task excludes other tasks from using the same buffer.

Therefore, current practice of moving data as early as possible is either: (i) unnecessary when viewed in isolation; or (ii) harmful when viewed in-the-large: due to finite capacity and competing transfers on the same link and buffer. Park & Humphrey [1] proposed a data throttle strategy so that when a task needs multiple input data sets, all of them arrive simultaneously. However, they do not discuss a mechanism for analysing and automatically deriving the throttling rates.

The benefit of using a data throttling strategy depends on the workflow structure, the intermediate data[1] size, and the execution environment (i.e. networking and computational resource characteristics, etc.). Such a strategy is applicable when a workflow task has multiple data inputs (we refer to these as "synchronisation points" in the workflow), enabling some input links to be throttled. We argue that the greater the number of such synchronisation points, the greater the benefit we are likely to see. We quantitatively measure this benefit, capturing it as a metric value that can be used to compare different workflow graph structures (achieving the same outcome). Our metric computation process converts the workflow DAG specification into a Petri net and combines this with intermediate data, computational, and network resource information. Subsequent analysis on the Petri net involves calculation of a "slack" (identifying which input data arrives first (the highest slack) and which arrives last (the slowest slack)). With this slack information, we calculate the value of the metric for each task and then aggregate this for the entire workflow. This paper is structured as follows: Section 2 describes related work in using Petri nets and automated management of data transfers in workflows. Section 3 describes how Petri nets can be used to derive performance analysis from workflow DAGs, leading to the description and calculation of our optimisation metric in Sections 4 and 5 – the main contribution of this paper. Section 6 includes a validation of the approach using the Montage workflow – leading to general conclusions that can be drawn outlined in Section 7.

2 Related Work

Petri nets and their extensions have been widely used for the specification, analysis and implementation of workflows [2]. Van der Aalst and Van Hee [3]

[1] Datasets obtained as a consequence of the execution of intermediate steps of a workflow are typically known as intermediate data.

describe WF-nets for modelling workflows with control operations such as sequence, choice, synchroniser, fork or merge. The types of structural analysis that can be undertaken includes correctness, deadlock analysis or liveness. In our approach a workflow graph (DAG) is mapped to a Petri net representation for subsequent analysis, as described in Section 3.

Data movement has also been a matter of study in a number of parallel and distributed research domains, an optimisation of MPI communications that both i) exploits spatial locality of files, for reducing the communications, and ii) dynamically chooses the best compression method for MPI messages is proposed in [4]. Yu and Buyya [5] classify automated data transfer strategies in workflows as centralised, mediated, and peer-to-peer. A centralised approach is often used when the time for data transfers is much smaller than computations. Taverna [6] typically utilises a centralised data transfer, due to the characteristics of the problems it tackles. In a mediated strategy the locations of intermediate data are managed by a distributed data management system. In a Peer-to-Peer (P2P) data is transferred directly between nodes, reducing both transmission time and the bottleneck caused by other approaches. We focus on data-intensive workflows using a P2P data sharing strategy. The Pegasus workflow system [7] supports both mediated and P2P transfers. In the mediated approach, Pegasus utilises a data replica catalogue that stores the intermediate data generated, so data can be subsequently retrieved rather than recomputed again. However, none of these approaches makes an effective usage of both network bandwidth and buffer space, often focusing instead on attempting to transfer the data as quickly as possible between workflow tasks. In [8], we proposed a method for deriving sub-optimal throttling values in data-intensive workflow DAGs.

3 Background: Petri Nets

A *Petri net* [9] (PN) is a mathematical formalism used for the modelling of concurrent and distributed systems. A PN is a bipartite graph, which consists of two set of nodes (places P, transitions T) and directed arcs. An arc connects a place with a transition, or vice versa, but it never connects a place with a place or a transition with a transition. The *initial marking* of place p, denoted as $\mathbf{m_0}(p)$, is the natural number of tokens that a place p may have. A *marking* $\mathbf{m}$ defines a distribution of tokens over the places. Normally, PNs have a graphical notation where places are represented by circles, transitions by bars, tokens by black dots and directed arcs by arrows. A transition of a PN is said to be *enabled* when all its input places (i.e., there exists an arc connecting a place with such a transition) have *tokens*. Once enabled, it can be *fired*. When fired, all these tokens are consumed (i.e., removed from input places) and placed in the output places. The firing of transitions of a PN is non-deterministic, i.e., when multiple transitions are enabled at the same time any of them can be fired.

The PN formalism can be extended by associating transitions with a delay. The delay is characterised by a random variable and represents the elapsed

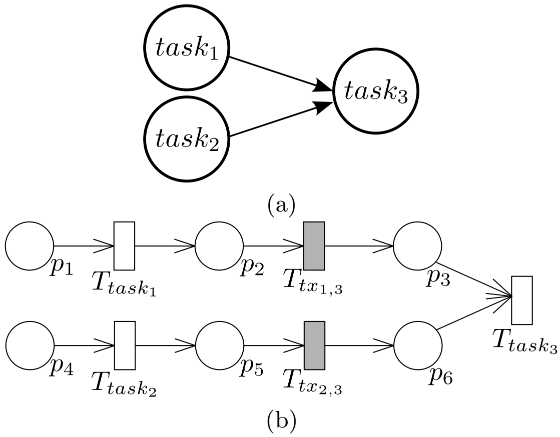


Fig. 1. (a) Workflow tasks and (b) its transformation to PN

time for firing the transition. Such an extension is called *Stochastic Petri nets* (SPNs) [10], where the random variables follow an exponential distribution. *Stochastic Marked Graphs* (SMGs) are a subset of SPNs, i.e., a SPN with a Marked Graph structure [9]: each place of the SPN has exactly one input and exactly one output arc.

In this paper, we make use of SMGs with exponential random distributions associated with transitions in order to model scientific workflows. In particular, we are interested in workflows expressed as a DAX, which is an XML representation of a DAG that most scientific workflow systems utilise. We automatically convert a DAX into a Petri net representation, in which vertices represent tasks and edges represent data dependencies between them. Fig. 1 depicts how we derive a PN model from a DAG. Fig. 1(a) shows three workflow tasks of a DAG, $task_1$, $task_2$ and $task_3$ and a data-link dependence from $task_1$ to $task_3$ and from $task_2$ to $task_3$, while Fig. 1(b) illustrates the transformation to a PN. Such a PN model fulfills the definition of a SMG model: there are timed transitions and each place has exactly one input and exactly one output arc.

Hence, each task of the workflow DAG is transformed to a place and a transition (represented by a white rectangle), joined by an arc. A task transmission is also transformed to a place and a transition (grey rectangle). For instance, $p_1 \rightarrow T_{task_1}$ represents $task_1$ of the workflow. Note that place p_1 models the input buffer of $task_1$. Finally, the data dependency between $task_1$ and $task_3$ is modelled by adding a place and a transition, p_2 and $T_{tx_{1,3}}$, respectively. Transition $T_{tx_{1,3}}$ represents the time spent in sending output data from $task_1$ to the input buffer of $task_3$ (place p_3). Note that transition T_{task_3}, which represents $task_3$, is not enabled until both places p_3 and p_6 have some token. Such places represent the synchronisation of $task_3$, which is unable to start its execution until it has received output data from $task_1$ and $task_2$.

4 Metric Definition

A task in a workflow DAG cannot start its execution until all its inputs are available. The strategy of receiving these input values as fast as possible is often not appropriate. As some inputs may arrive earlier than others, these inputs have to be buffered locally at the task, resulting in unnecessary use of buffer space. If such buffer space is a shared resource and of limited capacity, it remains blocked by the task, waiting for the remaining data to arrive. Hence, the greater the variation between arrival times of the different input data sets, the greater the inefficiency in buffer use. Intuitively, the objective of an effective data transfer is that each task with multiple inputs has all its data sets arrive simultaneously.

Our approach is therefore relevant for a workflow which has: (i) multiple synchronisation points (identified as tasks in the workflow containing more than one input, where all inputs are needed before the task can begin execution); (ii) difference in arrival times between the different inputs to such synchronisation point. The higher the value of (i) and (ii), the greater the possible optimisation we are likely to see with our approach. Both of these aspects depend on the structure of the workflow and the environment within which a workflow is enacted. Our approach could be used to re-write a workflow DAG that has a *structural imbalance*, i.e. a DAG containing multiple paths whose execution times differ significantly. Such an imbalance [1] may also arise due to a scheduler binding tasks to resources, faults or unexpected performance degradation, a slow network connection, or limited storage for the dataset.

In order to compute the metric, we convert the workflow DAG specification into a Petri net (as explained in Section 3), we subsequently feed the Petri net model with performance information on computational tasks, and network, as well as data size [8]. Petri net theory is subsequently used to analyse the Petri net model, and to obtain *slack* (μ) [11] values (a key concept in our analysis). Intuitively, a *slack* is a positive value associated with each input link to a synchronisation point and captures the time taken for an input data to be delivered to the sync. point. The higher the slack, the greater the probability that input delay will be higher, thereby delaying the execution of the task at the sync. point. A more formal description of a synchronisation point and the associated slack is as follows. Let workflow W be composed of a set of n tasks $T = \{t_1, \ldots, t_n\}$ where each task has m possible inputs, and where W is represented as a cyclic Petri net. A sync. point is defined as a task t_i with multiple inputs, i.e., $T' \subset T$ and $t_i \in T', m > 1$ – with the total number of sync. points being s, i.e., $|T'| = s, s \leq n$.

Let $d_{i,j}, j \in \{1 \ldots m\}$ represent the time taken for input j to arrive at task t_i. As each input arrives at different times, we can determine the value of $max(d_{i,j})$ for a sync. point t_i (i.e. the time taken for the slowest arriving input). Considering the entire workflow W, we can find the slowest path from the input to the output of the workflow, which also represents the workflow makespan – represented as $\sum_{i=1}^{n} max(d_{i,j}) + execution\ time(t_i), \forall_j$. The slack $\mu_{i,j} > 0, j \in \{1 \ldots m\}$, for input j of task t_i, can be calculated as:

$$\mu_{i,j} = \frac{(max_{j=1}^{m}(d_{i,j}) - d_{i,j})}{\sum_{i=1}^{n} max(d_{i,j}) + execution\ time(t_i), \forall_j} \tag{1}$$

The slack, therefore, is a measure of the time arrival of inputs with respect to the critical path of a workflow. Recall that the slack value for the slowest input $max(d_{i,j})$ is always 0. Our metric makes use of *slack* theory and incorporates both structural and execution environment aspects. The metric α can be expressed as:

$$\alpha = s \cdot \sum_{t_i \in T', j \in \{1 \ldots m\}} \mu_{i,j} \tag{2}$$

where s represents the number of synchronisation points in the workflow and $\sum_{t_i \in T', j \in \{1 \ldots m\}} \mu_{i,j}$, represents the summation of all the slacks that appear in the workflow. The value of α quantifies the potential benefit that a data-throttling strategy may achieve in comparison with a transmit as-fast-as-possible strategy. The metric indicates that the greater the number of tasks with multiple inputs, the greater the potential to save buffer/storage space. The second aspect the metric considers is the summation of all the slack values that appear in the workflow in case a transmit as-fast-as-possible strategy is applied. Ideally, an intelligent transfer strategy would make all the slack values equal to 0.

5 Metric Calculation

Our analysis algorithm and metric calculation is based on Little's law, $L = \lambda \cdot \chi$, which states that the average number of elements in a queue is given by the product of the system's throughput (λ), and the average time spent by the element in the queue (χ). In terms of a Petri net representation, each pair (input place, transition), which represents a pair (input, task) in a workflow, can be seen as a simple queueing system for which Little's law can be directly applied: $\mathbf{m}(p) = \Theta \cdot \delta_t + \mu(p)$, where $\mathbf{m}(p)$ denotes the average number of tokens in place p, Θ represents the system's throughput, δ_t is the average delay associated for the computation of transition t (it models a task), and $\mu(p)$ is the slack value. It should be noted that Little's average queueing time is the addition of the average waiting time due to a possible synchronisation and the average service time, which in this case is δ_t.

In consequence, the first step in the analysis algorithm is to compute the system's throughput Θ, the inverse of the time delay of the critical path. The time delay for the critical path is obtained by solving a Linear Programming

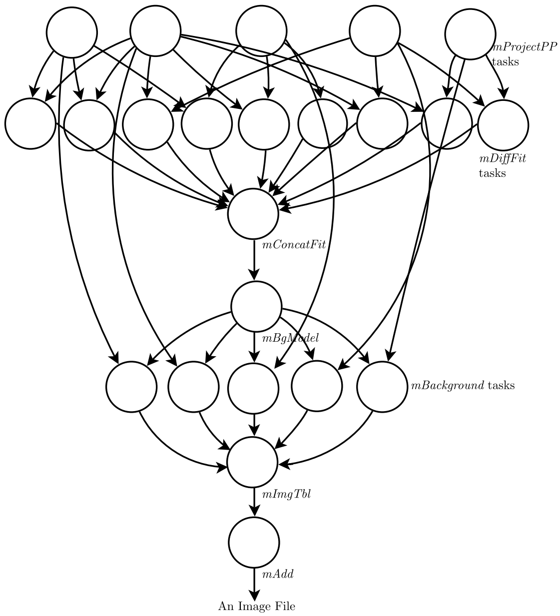


Fig. 2. Montage workflow graph with 25 tasks

Problem (LPP) [12]. The second step is to obtain a special marking (a PN state) of the SMG, which is called *tight marking*, denoted as $\tilde{\mathbf{m}}$. This state fulfils a number of properties, namely i) the tight marking is reachable from the initial marking (initial state), ii) for each input place p, $\tilde{\mathbf{m}}(p) \geq \Theta \cdot \delta$, which is related to Little's Law, and iii) for each transition t, there is at least one input that has no slack, $\tilde{\mathbf{m}}(p) = \Theta \cdot \delta_t$. The tight marking is also computed by a LPP. Finally, for all place p, the slack is easily computed by conducting this algebraic operation, $\mu(p) = \tilde{\mathbf{m}}(p) - \delta_t \cdot \Theta$.

6 Evaluation

In order to evaluate how our metric α measures the effectiveness of utilising a data throttling strategy, we conducted several experiments using the Montage [13] workflow. This workflow is used for creating image mosaics in astrophysics (using data from different scientific instruments) and a specification of Montage structure can be found in Fig. 2. The structure of the workflow is quite regular, therefore the workflow imbalance, if any, must be due to the execution environment. However, Montage workflow structure depends on the number of input files to assemble. We considered 3 versions of Montage, with 25, 50 and 100 tasks, and having each workflow 5, 10 and 100 input files, respectively. A DAX description of this workflow, along with performance information collected from past executions is available at the Pegasus workflow system [7] Web site[2].

[2] https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator

We have simulated the enactment of these 3 workflows by making use of the SimGrid [14] tool. We assumed an environment with the number of machines being the same as the number of nodes in the workflow. As for the network topology, we assumed a single output data-link per host, a network bandwidth of 100Mbps and a latency of 10^{-4}s.

Table 1. Mean & standard deviation values of buffer waiting time for Montage workflow for $25, 50, 100$ tasks

Montage task	25 tasks		50 tasks		100 tasks	
	Mean	σ	**Mean**	σ	**Mean**	σ
mDiffFit	$1.7726s$	$1.1800s$	$1.5739s$	$1.2044s$	$1.9819s$	$1.6399s$
mConcatFit	$15.9589s$	–	$57.5424s$	–	$226.1896s$	–
mBackground	$15.0307s$	$1.1194s$	$19.0069s$	$1.4327s$	$25.1063s$	$1.8597s$
mImgTbl	$2.9567s$	–	$1.4934s$	–	$4.0822s$	–

For each Montage workflow, Table 1 shows the mean values and standard deviation, σ, of the buffer waiting time of the Montage tasks with multiple input data, namely *mDiffFit*, *mConcatFit*, *mBackground*, and *mImgTbl*. A buffer waiting time was computed as the elapsed time between the arrival of the last and the first input. The results show that for some tasks, the higher the number of synchronisation points, the longer the buffer wait times. The main reason for this is that the synchronisation points of larger Montage workflows have a larger number of inputs in comparison with smaller sized workflows. In particular, this fact can be observed in task *mConcatFit*.

Table 2. Metric values computed for the considered Montage workflows

No. tasks	**No. tasks with inputs**	$\sum \mu$	**Metric** α
25	15	1.5173	22.7591
50	32	2.7681	88.5785
100	75	5.3371	400.2842

The metric values for each considered Montage workflow are shown in Table 2. The first column indicates the number of total tasks in the workflow, while the second shows the number of tasks with multiple input dependencies. Finally, the third column, $\sum \mu$, represents the summation of slacks (computed as indicated in Section 5 with the MATLAB LP toolkit) and the last column presents the value of the α metric computed. As it can be expected, and it is remarked by the buffer waiting time given in Table 1, the more value of α, the better expected reduction of waiting time by using a data-throttling strategy. For the Montage workflow, it can be then concluded that the higher the number of task in the Montage version, the more important a data throttling strategy will be, so that buffer occupancy can be utilised more effectively.

7 Conclusions

The enactment of data intensive workflows often involves the transfer of files between tasks. Where a task requires multiple files to execute, it must remain idle until all files are available. Hence, an effective and intelligent data transfer strategy ideally would consist in throttling data, so that all input files for a given task arrive simultaneously. In this paper, we study the potential impact that applying such an strategy can have on a workflow, expressed as a Directed Acyclic Graph (DAG). We propose a *quantitative* metric based on the workflow structure, and on performance information derived from past historical executions. We convert the DAG specification into a Petri net model, feed it with such a performance information, and conduct analysis over it to obtain task inputs with *slack*. A slack is a positive value that when computed, appears at a task input that is likely to be idle, waiting for other datasets to arrive. The higher the slack value, the more likely the associated input will have to wait for a longer period. Our metric is proportional to the number of synchronisation points in the workflow and to the sum of all slack values appearing in the workflow. This metric is intended for use at the design-stage, to compare various DAG structures and evaluate their *potential* for optimisation (of network bandwidth and buffer use).

We conducted experiments over 3 Montage workflows, with different sizes. For each synchronisation point in the workflow, we measured the buffer occupancy time and obtained our metric. We observed that the higher the value of the metric, the higher the buffer occupancy time within the Montage workflows. The throttling of data becomes more important with the growing size of the Montage workflow. Due to its characteristics, a Montage workflow with a larger number of tasks suffers from larger buffer occupancy, and this will require more buffering space for the inputs. Although demonstrated through a single workflow our approach is quite general in scope and can be applied to any workflow described using the DAX representation. As future work, we are considering the design and implementation of a tool for workflow design that assists users in their workflow configurations, and helps them determine: i) comparison of different workflows / workflow configurations for potential optimisation, ii) sub-optimal throttling values, iii) performance speed-up or degradation when applying data throttling, and iv) buffer storage needs.

Acknowledgements. This work was supported by the Projects EU FP7 DISC (INFSO-ICT-224498), Spanish CICYT DPI2010-20413, and TIN2010-17905, and by Fundación Aragón I+D. Mr Rodríguez's work was supported by grants from DGA (CONAID) and CAI, "Programa Europa XXI" ref. n. IT 5/11, and Ministerio de Educación, Cultura y Deporte, ref n. TME2011-00255.

References

1. Park, S.M., Humphrey, M.: Data Throttling for Data-Intensive Workflows. In: IEEE International Symposium on Parallel and Distributed Processing, pp. 1–11 (April 2008)
2. van der Aalst, W., van Hee, K.: Workflow Management: Models, Methods, and Systems. MIT Press Books, vol. 1. The MIT Press (2004)
3. van der Aalst, W.M.P., Hirnschall, A., Verbeek, H.M.W.: An Alternative Way to Analyze Workflow Graphs. In: Pidduck, A.B., Mylopoulos, J., Woo, C.C., Ozsu, M.T. (eds.) CAiSE 2002. LNCS, vol. 2348, pp. 535–552. Springer, Heidelberg (2002)
4. Filgueira, R., Carretero, J., Singh, D.E., Calderón, A., Nuñez, A.: Dynamic-compi: dynamic optimization techniques for mpi parallel applications. The Journal of Supercomputing 59(1), 361–391 (2012)
5. Yu, J., Buyya, R.: A Taxonomy of Workflow Management Systems for Grid Computing. CoRR 34(3), 44–49 (2005)
6. Oinn, T., Greenwood, M., Addis, M., Alpdemir, M.N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M.R., Senger, M., Stevens, R., Wipat, A., Wroe, C.: Taverna: lessons in creating a workflow environment for the life sciences: Research Articles. Concurr. Comput.: Pract. Exper. 18(10), 1067–1100 (2006)
7. Deelman, E., Mehta, G., Singh, G., Su, M., Vahi, K.: Pegasus: Mapping Large-Scale Workflows to Distributed Resources. In: Workflows for eScience, pp. 376–394. Springer (2007)
8. Rodríguez, R.J., Tolosana-Calasanz, R., Rana, O.F.: Automating Data-Throttling Analysis for Data-Intensive Workflows. In: Proceedings of CCGrid (accepted for publication, 2012)
9. Murata, T.: Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE 77, 541–580 (1989)
10. Molloy, M.: Performance Analysis Using Stochastic Petri Nets. IEEE Transactions on Computers C-31(9), 913–917 (1982)
11. Rodríguez, R.J., Júlvez, J.: Accurate Performance Estimation for Stochastic Marked Graphs by Bottleneck Regrowing. In: Aldini, A., Bernardo, M., Bononi, L., Cortellessa, V. (eds.) EPEW 2010. LNCS, vol. 6342, pp. 175–190. Springer, Heidelberg (2010)
12. Campos, J., Silva, M.: Embedded Product-Form Queueing Networks and the Improvement of Performance Bounds for Petri Net Systems. Performance Evaluation 18(1), 3–19 (1993)
13. Berriman, G.B., Deelman, E., Good, J., Jacob, J.C., Katz, D.S., Laity, A.C., Prince, T.A., Singh, G., Su, M.H.: Generating Complex Astronomy Workflows. In: Taylor, I.J., Deelman, E., Gannon, D.B., Shields, M. (eds.) Workflows for e-Science, pp. 19–38. Springer, London (2007)
14. Casanova, H., Legrand, A., Quinson, M.: SimGrid: a Generic Framework for Large-Scale Distributed Experiments. In: 10th IEEE International Conference on Computer Modeling and Simulation (March 2008)

Goal, Video Description and Capability Ontologies for Fish4Knowledge Domain*

Gayathri Nadarajan and Yun-Heh Chen-Burger

School of Informatics, University of Edinburgh, U.K.
gaya.n@ed.ac.uk, jessicac@inf.ed.ac.uk

Abstract. We created a set of domain ontologies that are based on user requirements for the Fish4Knowledge (F4K) project – goal, video description and capability. The roles of the ontologies are to 1) support the development of appropriate functions of the project's workflow system, and 2) serve as a communication media to interface with other F4K components. The ontologies were designed with collaboration with image processing experts, marine biologists and user interface experts to capture the domain knowledge succinctly. They were utilised in the first version of our workflow composition and execution system for video classification, fish detection and counting tasks. They will continue to evolve with F4K's needs and are envisaged to interface with other components.

Keywords: ontologies, semantics based workflows, requirements based virtual workflow system, intelligent video processing.

1 Introduction

The workflow component of the Fish4Knowledge (F4K) project is responsible for the composition and execution of a set of video and image processing (VIP) modules on high performance computing machines based on user requirements and descriptions of the video data. The workflow component interprets the user requirements as high level VIP tasks, create workflows based on the procedural constraints of the VIP modules to ultimately invoke and manage their execution in a distributed environment. In our intelligent workflow system, we have chosen to use an ontological-based approach to guide the automatic generation of a "virtual workflow machine" based on a set of closely related ontologies. This allows a separation between the problem and application descriptions and the workflow mechanism. As a result, the workflow machine may work in different problem domains if the problem and application descriptions are changed. Consequently, this will promote reusability and provide a conceptualisation that can be used between different domain experts, such as marine biologists, image processing experts and workflow engineers. These ontologies are also pivotal for reasoning. For instance, in the selection of optimal VIP software modules, the Capability Ontology is used to record known heuristics obtained from VIP experts.

* This research was funded by European Commission FP7 grant 257024, in the Fish4Knowledge project (www.fish4knowledge.eu).

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 154–163, 2012.

This paper describes our efforts in creating a set of suitable domain ontologies that are based on user requirements (from marine biologists) for our intelligent workflow system, the ontologies will 1) support the development of appropriate functions of the workflow system, and 2) serve as a communication media to interface with other F4K components. We first describe the 20 scientific questions provided by F4K's marine biologists, and then provide an analysis of these questions from the workflow system's point of view – that is based on the capabilities of the VIP modules available to us (Section 2). Based on a mapping between the user requirements and a high level abstraction of the capabilities of the VIP modules, we have constructed the **Goal Ontology** (Section 3). The **Video Description Ontology** (Section 4) contains the environmental factors related to the videos, among others. The **Capability Ontology** (Section 5) describes details of the VIP tools and techniques. For ontology development and visualisation purposes, OWL 1.0 [3] was generated using Protege version 4.0. Where applicable, ontological diagrams were derived using the OntoViz plugin.

To date, the Goal Ontology contains 52 classes, 85 instances and 1 property, the Video Description Ontology has 24 classes, 30 instances and 4 properties and the Capability Ontology has been populated with 42 classes, 71 instances and 2 properties. They have supported the first version of the workflow system that has been evaluated for efficiency, adaptability and user learnability in video classification, fish detection and counting tasks [4,5].

2 Related Work

Ontologies have been used for describing VIP domain knowledge by several major projects, such as aceMedia [1], VIDI-Video [2], and NeOn [6], just to name a few. VIDI-Video and aceMedia have worked on developing multimedia ontologies for video annotation tasks. The aceMedia project has been a successful effort in utilising ontologies to capture high level descriptions to guide the low level technical tasks of multimedia annotation via their core, domain and multimedia ontologies, all described in RDFS. The VIDI-Video project uses semantics-based and machine learning technologies to explore the area of semantics-based video search. The NeOn project has developed a network of fisheries ontologies that has some relevance to our work. We have identified and intergrated FAO's concepts and properties that were relevant to F4K. These include fishing areas[1], land areas[2] and biological entities[3]. This effort is the first step in integrating our work with existing standards in the ontological field.

3 Goal Ontology

The Goal Ontology contains the high level questions posed by the user, interpreted by the system as VIP tasks (termed as goals) and the constraints to the

1 http://www.fao.org/fishery/cwp/handbook/H/en
2 http://www.fao.org/countryprofiles/geoinfo.asp
3 http://www.fao.org/fishery/collection/asfis/en

Table 1. List of 20 scientific questions posed by F4K's Taiwanese marine biologists

Q1	How many species appears and their abundance and body size in day and night including sunrise and sunset period.
Q2	How many species appears and their abundance and body size in certain period of time (day, week, month, season or year). Species composition change within one period.
Q3	Give the rank of above species, *i.e.* listed according to their abundance or dominance. How many percent are dominant (abundant), common, occasional and rare species.
Q4	Fish color pattern change and their behavior in the night for diurnal fish and vice versa for nocturnal fishes.
Q5	Fish activity within one day (24 hours).
Q6	Feeding, predator-prey, territorial, reproduction (mating, spawning or nursing) or other social or interaction behavior of various species.
Q7	Growth rate of certain species for a certain colony or group of observed fishes.
Q8	Population size change for certain species with one period of time.
Q9	The relationship of above population size change or species composition change with environmental factors, such as turbidity, current velocity, water temperature, salinity, typhoon, surge or wave, pollution or other human impact or disturbance *etc.*
Q10	Immigration or emigration rate of one group of fishes inside one monitoring station or one coral head.
Q11	Solitary, pairing or schooling behavior of fishes.
Q12	Settle down time or recruitment season, body size and abundance for various fishes.
Q13	In certain area or geographical region, how many species could be identified or recognized easily and how many species are difficult. The most important diagnostic character to distinguish some similar or sibling species.
Q14	Association among different fish species or fish-invertebrates.
Q15	Short term, mid-term or long term fish assemblage fluctuation at one monitoring station or comparison between experimental and control (MPA) station.
Q16	Comparison of the different study result between using diving observation or underwater real time video monitoring techniques. Or the advantage and disadvantage of using this new technique.
Q17	The difference of using different camera lens and their angle width.
Q18	Is it possible to do the same monitoring in the evening time.
Q19	How to clean the lens and solve the biofouling problem.
Q20	Hardware and information technique problem and the possible improvement based on current technology development and how much cost they are.

goals. Before it could be constructed, a list of queries posed by marine biologists were examined. These questions are listed in Table 1.

As a starting point, the 20 questions provided by the marine biologists were mapped to the VIP tasks (goals) that they fall into based on the types of tasks that are required to be carried out to address these questions. A VIP task as indicated here is a high level goal in our Goal Ontology that can be understood by an image processing expert, who writes low level programs or software to solve it. Table 2 contains the result of the analysis after communication with image processing experts, marine biologists and user interface experts. The queries are indicated as Q1 to Q20 and the eight main VIP goals are contained in columns 2 to 9 of the table.

Table 2. Mapping of high-level VIP tasks to the 20 questions as provided by Fish4Knowledge's marine biologists

Query	Fish Detection and Tracking	Population Related Analysis	Fish Species Classification	Behaviour Understanding	Fish Clustering	Fish Feature Analysis	Event Detection	Aggregated Analysis	Assumptions/Definitions
1	x	x	x			x			Assumption: Sunrise & sunset are defined by IP teams as a 10-minute period when videos change from dark to colour & vice-versa. Night videos are not available.
2	x	x	x					x	Definition: *Species composition* is defined as the number of fish from each species.
3	x	x	x	(x)		x		x	
4	x	x	x	x				x	Comment: *Diurnal* and *nocturnal* fishes will need to be determined manually by F4K researchers and populated in the database. Colour pattern change may not be possible to be performed as the change will occur within 1–2 seconds, which is too quick for current VIP techniques to detect.
5	x	x	x	x		x			Assumption: *Fish activity* refers to activity of *all* the fish in a video clip.
6	x	x	x	x		x		(x)	Comment: Territorial behaviour is seen when a fish species attack other fish or human entering the area that they are in.
7	x	x	x			x		x	Definition: *Fish group* is a group of fish (of the same or closely related species) living together in one area. Definition: *Fish colony* refers to the same fish species. Definition: Growth rate = birth rate – death rate + immigration rate – emigration rate
8	x	x	x		(x)	x	(x)	x	Assumption: *Population size change* depends on (not exclusively, but most commonly) immigration, emigration, birth rate and death rate.
9	x	x	x	(x)		x	x	x	Comment: NCHC could provide water salinity, temperature, turbidity, ph value and dissoved oxygen. In the future typhoon incidence is a possibility.
10	x	x	(x)			(x)		x	Comment: Mathematical formula for *immigration* and *emigration rate* are required for aggregated analysis.
11	x	x	x	x		x			
12	x	x	x			x		x	Definition: *Settle down time* refers to the time it takes to colonise a new area through immigration. In marine terms, settlement refers to the colonisation of a previously unavailable area, *e.g.* new structure

Table 2. *(continued)*

Query	Fish Detection and Tracking	Population Related Analysis	Fish Species Classification	Behaviour Understanding	Fish Clustering	Fish Feature Analysis	Event Detection	Aggregated Analysis	Assumptions/Definitions
									(sunken ship) or an area that has been cleared (usually by a destructive event, *e.g.* storms, human action). Definition: *Recruitment season* is the phenomenon where new born fishes survive and are added to the overall population.
13	x	x	x		x	x		x	Comment: Similar/sibling species could be physically similar (*e.g.* shape, size, number of spikes), or having the same genus but not species. Comment: This task requires labelled data, *i.e.* ground truth in order to determine whether a fish species is easy or difficult to be recognised.
14	x	x	x	x	(x)	x		(x)	Definition: *Association* refers to the co-occurrence of species. It could be considered with the number of co-occurrences per associated species, average time-span and threshold of time-span between two occurrences. An example is the symbiotic relationship between fish and fish-invertebrates.
15	x	x	x	(x)		x		x	Definition: A *fish assemblage* is a group of interacting populations within a specific area. They must interact *e.g.* predator/prey interactions, compete for food, space or other resources. Fish assemblage fluctuation is the change within the structure of the assemblage, *e.g.* increase/decrease in numbers either either overall or within individual populations.
16								x	Comment: This task requires results from manual processing in order for comparisons to be performed.
17								x	Comment: This is not possible at present.
18								x	Comment: This is not possible at present unless infra-red cameras are installed and recordings are provided.
19							x	x	Comment: This question should be rephrased as 'When to clean the lens'. How to clean the lens and solve the bio-fouling problem is beyond F4K's IP capabilites.
20								x	Comment: This task is not possible at present.

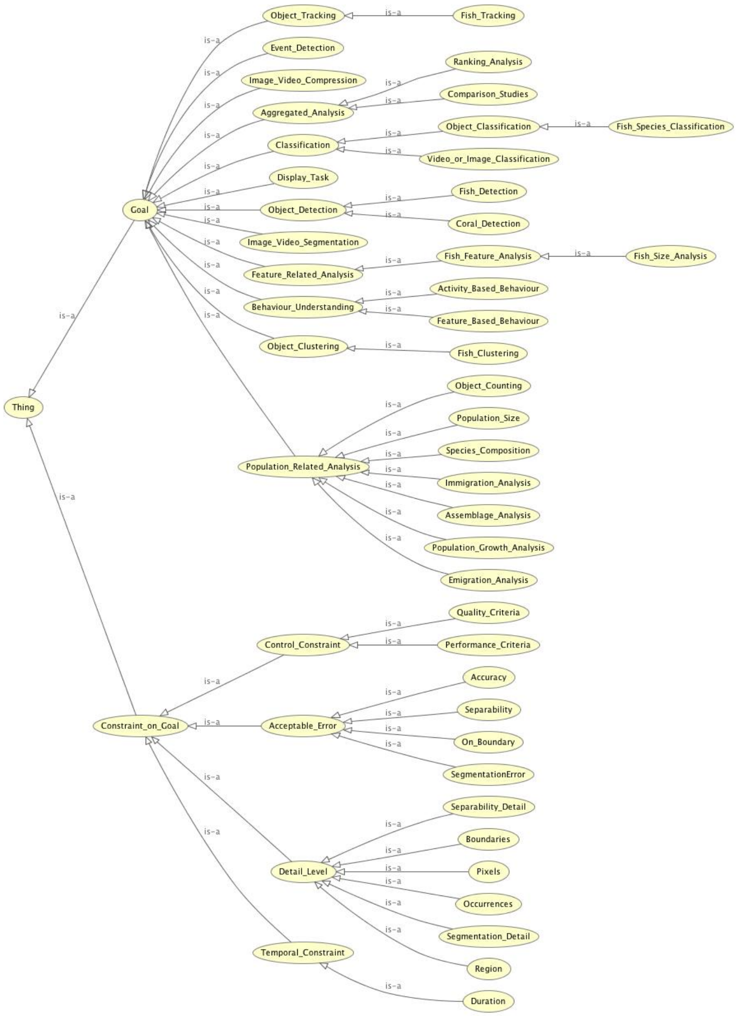


Fig. 1. Goal ontology denoting the main classes of goals and constraints that were applicable to the 20 questions posed by Taiwanese marine biologists

For each query the relevant VIP task(s) that can be used to support it are marked with an ‘x’. An ‘x’ marked in parentheses indicate that the particular VIP task could be associated with the query, but it could not be determined with absolute certainty that this function will be supported within this project.

This may be due to technical reasons or the time constraints imposed on this project. The VIP tasks identified here constitute the main goals that we describe and use in the Goal Ontology. Along with these main goals, we add higher level goals to provide a classification structure and linkage to possible user queries. Fig. 1 depicts the Goal Ontology pictorially. The main relationship between classes shown here is the sub-class relationship, *is-a*. Instances and properties are not shown due to the limitation of the visualisation tool.

The 'Goal' class is the umbrella of concept that includes the VIP tasks identified in Table 2. The eight main VIP tasks can be found in Figure 1. We also created a level of intermediate classes between the goal class and the lower level VIP goals to increase the flexibility and readability of the ontology. Under these general concepts, more specific goals may be defined, for example 'Fish Detection', 'Fish Tracking', 'Fish Clustering', 'Fish Species Classification' and 'Fish Size Analysis'. The principle behind keeping the top level concepts more general is also to allow the ontology to be easily extended to include other (new) tasks as appropriate as the project develops.

'Constraint on Goal' refers to the conditions that restrict the video and image processing tasks or goals further. In F4K's context, the main constraint for a VIP goal is the 'Duration', a subclass of 'Temporal Constraint'. Each task may be performed on all the historical videos, or a portion specified by the user – within a day, night, week, month, year, season, sunrise or sunset (all specified as instances of the class 'Duration'). Other constraints types include 'Control Constraint', 'Acceptable Error' and 'Detail Level'. The control constraints are those related to the speed of VIP processing and the quality of the results expected by the user. Acceptable errors are the threshold for errors that the user may want to insert. An example of this is the criterion 'Accuracy', which states the accuracy level of a detected object. The class 'Detail Level' contains constraints that are specific to particular details, for example detail of 'Occurrence' is used for detection tasks to constrict the number of objects to be detected.

The Goal Ontology is used for consistency checks when a user query is detected in the system. It can check that the query matches with a goal or set of goals that is achievable within the workflow system. It is also used to guide the selection of higher level tasks for workflow and formulate input values to the reasoning engine that is responsible for searching the VIP solution set for a VIP task, *i.e.* to compose the workflow.

4 Video Description Ontology

The Video Description Ontology describes the concepts and relationships of the video and image data, such as what constitutes video/image data, the acquisition conditions such as lighting conditions, colour information, texture, **environmental conditions** as well as spatial relations and the range and type of their values. Fig. 2 gives a pictorial overview of the main components of the Video Description Ontology. The upper level classes include 'Video Description', 'Descriptor Value', 'Relation', and 'Measurement Unit'.

The main class of this ontology is the 'Video Description' class, which has two subclasses – 'Description Element' and 'Descriptor'. A description element can be either a 'Visual Primitive' or an 'Acquisition Effect'. A visual primitive describes visual effects of a video/image such as observed object's geometric and shape features, *e.g.* size, position and orientation while acquisition effect descriptor contains the non-visual effects of the whole video/image that contains the video/image class such as the brightness (luminosity), hue and noise conditions.

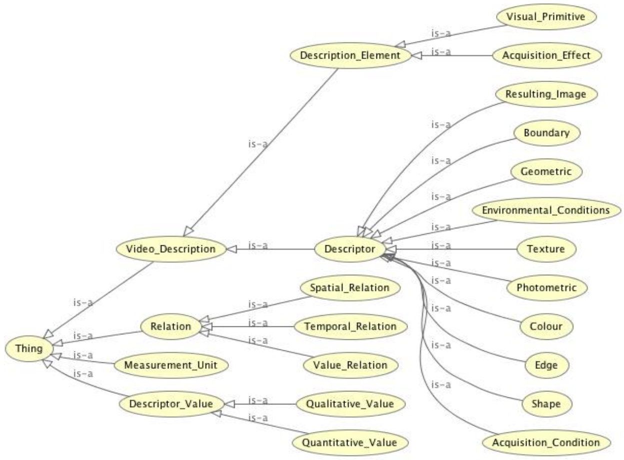


Fig. 2. Main concepts of the Video Description Ontology

The descriptor for the description elements are contained under the 'Descriptor' class and are connected to the 'Description Element' class via the object property 'hasDescriptionElement'. Typical descriptors include shape, edge, colour, texture and environmental conditions. Environmental conditions, which are acquisitional effects, include factors such as current velocity, pollution level, water salinity, surge or wave, water turbidity, water temperature and typhoon.

These values that the descriptors can hold are specified in the 'Descriptor Value' class and connected by the object property 'hasValue'. For the most part, qualitative values such as 'low', 'medium' and 'high' are preferred to quantitative ones (*e.g.* numerical values). 'Qualitative' values could be transformed to quantitative values using the 'convertTo' relation. This would require the specific measurement unit derived from one of the classes under the concept 'Measurement Unit' and conversion function for the respective descriptor *e.g.* a low velocity could be interpreted as movement with velocity within a range of 0 and $25ms^{-}1$[4]. Some descriptor values can be tied to their appropriate measurement units. The property that specifies this is 'hasMeasurementUnit', which relates instances in the class 'Descriptor' to instances in the class 'Measurement Unit'. This ontology can be used to describe the videos and external effects on it such as environmental conditions. As with the Goal Ontology it is used for consistency checking when a set of video descriptors are supplied to the system from the user interface.

5 Capability Ontology

The Capability Ontology (Fig. 3) contains the classes of video and image processing tools, techniques and performance measures of the tools with known

[4] Currently, there is no fixed conversion formula. The actual conversion formula used will be determined as we gain more experience using our workflow system over time.

Fig. 3. Main concepts of the Capability Ontology

domain heuristics. This ontology will be used to identify the tools that will be used for workflow composition and execution of VIP tasks. In our context, it is used by a reasoner for the selection of optimal VIP tools. The main concepts intended for this ontology have been identified as 'VIP Tool', 'VIP Technique' and 'Domain Descriptions for VIP Tools'. Each VIP technique can be used in association with one or more VIP tools. A VIP tool is a software component that can perform a VIP task independently, or a function within an integrated vision library that may be invoked with given parameters. 'Domain Description for VIP Tool' represent a combination of known domain descriptions (video descriptions and/or constraints to goals) that are recommended for a subset of the tools. This will be used to indicate the suitability of a VIP tool when a given set of domain conditions hold at a certain point of execution. At present these domain descriptions are represented as strings and tied to VIP tools, *e.g.* Gaussian background model would have the description 'Clear and Fast Background Movement' to indicate the best domain scenario for it to be selected.

The main types of VIP tools are video analysis tools, image enhancement tools, clustering tools, image transform tools, basic structures and operations tools, object description tools, structural analysis tools and object recognition and classification tools. At present fish detection and tracking have been performed more than the other tasks within Fish4Knowledge. Hence the ontology has been populated with most of the tools associated with these tasks. For other tasks that have not been performed, *e.g.* fish clustering, the ontology will be extended and populated in due course. Detection and tracking tools fall under the class 'Video Analysis Tool'. Other types of video analysis tools are event detection tools, background modelling tools and motion estimation tools.

The class 'Object Description Tool' specifies tools that extract features such as colour, texture, size and contour, while image transform tools are those concerned with operations such as point, geometric and domain transformations. 'VIP Technique' is a class that contains technologies that can perform VIP operations. For now, two types of machine learning techniques have been identified. These techniques could be used to accomplish the task of one or more VIP tools. For example, neural netwworks can be used as classifiers as well as detectors.

The Capability Ontology can be used for reasoning during workflow composition using planning. As planning takes into account preconditions before selecting a step or tool, it will assess the domain conditions that hold to be used in conjunction with an appropirate VIP tool.

6 Conclusions

We have presented three ontologies that relate to the goals, capabilities and environmental conditions within Fish4Knowledge. Development of these ontologies act as a starting point to bring various expertise within the project together (*e.g.* image processing, workflow, marine science and user interface). At its current status, these ontologies have been used to guide the construction of VIP workflows for single videos on single CPU machines. New workflows will be handling multiple videos on a high performance computing environment, which would require appropriate prioritising, scheduling, monitoring and fault tolerance management strategies. At present the ontologies are interfacing the workflow and the image processing components, it is envisaged that the ontologies would also serve as a communication media to interface with other F4K components in due course.

References

1. aceMedia: Integrating Knowledge, Semantics and Content for User-Centred Intelligent Media Services (2004), http://www.acemedia.org/aceMedia
2. Bertini, M., Bimbo, A.D., Serra, G., Torniai, C., Cucchiara, R., Grana, C., Vezzani, R.: Dynamic Pictorially Enriched Ontologies for Digital Video Libraries. IEEE MultiMedia 16(2), 42–51 (2009)
3. McGuinness, D., van Harmelen, F.: OWL Web Ontology Language. World Wide Web Consortium (W3C) (2004), http://www.w3.org/TR/owl-features
4. Nadarajan, G., Chen-Burger, Y.H., Fisher, R.B.: A Knowledge-Based Planner for Processing Unconstrained Underwater Videos. In: IJCAI 2009 Workshop on Learning Structural Knowledge From Observations (STRUCK 2009) (2009)
5. Nadarajan, G., Chen-Burger, Y.-H., Fisher, R.B.: SWAV: Semantics-Based Workflows for Automatic Video Analysis. In: O'Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS, vol. 6682, pp. 681–691. Springer, Heidelberg (2011)
6. NeOn: Networked Ontologies Project (2006-2010), http://www.neon-project.org

An Early Comparison of Commercial and Open-Source Cloud Platforms for Scientific Environments

Ivan Voras, Marin Orlić, and Branko Mihaljević

University of Zagreb Faculty of Electrical Engineering and Computing, Zagreb, Croatia
{ivan.voras,marin.orlic,branko.mihaljevic}@fer.hr

Abstract. Cloud computing promises efficient use of hardware resources through virtualization and elastic computing facilities. Various cloud computing solutions have emerged on the market from open-source communities and commercial vendors. In this paper we discuss criteria for feature comparison of private cloud platforms and compare several open-source and commercial products. We test performance of hypervisors used in these clouds with a set of benchmark suites containing tests for various aspects of the system. We discuss the results in the context of what is commonly described as a scientific workload. The described feature and performance differences can help make wiser platform choices.

Keywords: Cloud-computing, open-source, commercial/closed, hypervisors.

1 Introduction

Cloud computing is a combination of technologies for resource management and provisioning with the goals of achieving cost control, elasticity, ease of use and mass deployment. In the wake of the cloud trend, numerous open-source and commercial cloud products have emerged. In this work, we focus on private-cloud infrastructure-as-a-service (IaaS) products, and present a comparison of features and performance of open-source and commercial solutions. We look at private clouds as primary platforms in scientific environments, with full local control over the platform.

Open-source solutions are often the first choice in scientific environments for their lower initial investments and support for scientific libraries and tools, as well as openness to customizations. Aggressive pricing and licensing options from vendors of commercial products leads us to expect a variety of cloud environments managing mixed virtualizations in the future. Previous performance comparisons of cloud services for scientific computing focused on open-source and modified open-source solutions [1]. In this paper, we compare performance of open-source and commercial hypervisors, the key elements and enablers of cloud platforms.

The rest of this paper is organized as follows. Sections 2 and 3 describe select open-source and commercial cloud platforms, respectively. Section 4 describes the evaluation criteria used to compare product features and presents the comparison results. Section 5

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 164–173, 2012.

compares the performance of open-source and commercial hypervisors, and Section 6 concludes the paper and discusses the future work in this area.

2 Open-Source Cloud Platforms

Open-source cloud platforms employ open-source hypervisors (KVM and Xen), but some of them also support commercial/closed hypervisors with exposed interfaces (VMware). Cloud platforms combine various tools of the underlying OS and virtualization layer with their own components in a more or less seamless cloud interface.

Eucalyptus Community Cloud. Eucalyptus is one of the most adopted cloud computing architectures, with open-source (Community Cloud) and commercial versions (Enterprise Edition). ECC is a flexible and highly modular system with components exposed in the form of web services interoperable with the Amazon AWS API which allows seamless integration with existing Amazon public cloud services. ECC currently supports Xen and KVM virtualizations and can be deployed on all major Linux distributions (Ubuntu, CentOS, Debian, RHEL, openSUSE, SLES etc.).

OpenNebula. OpenNebula is a management toolkit for private and public clouds, which orchestrates existing systems and services, relying on Linux and external products for virtualization, network, storage or security technologies. Infrastructure abstraction and modular approach support standardization and interoperability with most common virtualizations (Xen, VMware, and KVM), interfaces (Amazon AWS, VMware vCloud, and OGF OCCI), and APIs. Its features include secure management of virtual images, machines, networks and storage, authentication, multi-tenancy, quota management, and cloud-bursting with Amazon EC2.

3 Commercial Cloud Platforms

We have chosen the following commercial platforms for this discussion: VMware's (ESXi hypervisor), Microsoft's (Hyper-V hypervisor), and Citrix's (XenServer hypervisor, a modified version of the open-source product Xen). These hypervisors are a basis for other commercial offerings, especially those coming from OEMs[1] , and benchmark results also apply for those products.

VMware vCloud. VMware's offering is the oldest and the most feature-complete, relying on the popularity and proven track record of its virtualization products (ESXi, vSphere). On top of the virtualization platform, a comprehensive management platform was built and subsequently extended for use in cloud-like environments with the vCloud Director product. Though VMware's offer is truly complete, it is also somewhat fragmented, with services such as security management (vShield), billing

[1] HP CloudSystem Matrix, `http://h18004.www1.hp.com/products/blades/components/matrix/main.html`

(Chargeback Manager), high availability (Server Heartbeat), and others delivered and charged separately, which may significantly increase initially projected costs.

As an independent vendor not tied to specific hardware platforms or operating systems, VMware has created the most platform-agnostic product among the evaluated solutions, which has been adopted by hardware vendors and which fully supports the widest number of guest operating systems among the evaluated solutions.

Microsoft Private Cloud. Microsoft's presence as a server virtualization and cloud computing vendor is very recent, beginning in 2008 with the introduction of its Hyper-V hypervisor product. The features and abilities of Hyper-V are noticeably sub-par when compared to others we have evaluated, lacking important features such as live migration, support for guest OS booting from SCSI drives, memory over-commit, support for more than 4 virtual CPUs, and official support for popular non-Windows operating systems. On the other hand, Hyper-V is tightly integrated with other Microsoft's products and delivers an environment which is easy to use and manage, especially when virtualizing Windows workloads. The “Private cloud” solution integrates several products which automate resource management (most notably, the System Center Virtual Machine Manager).

Citrix CloudStack. Citrix's cloud solution is centered on its XenServer product, and supported by the infrastructure and the community of the CloudStack open source project, whose parent company (Cloud.com) it recently acquired. The relationship between Citrix and CloudStack seems to be somewhat complex, with Citrix supporting and endorsing it for its cloud computing efforts, as well as selling “Enterprise” and “Service Provider” editions. Of all the evaluated solutions, Citrix's is perhaps the most on the border between being a truly open and a commercial/closed solution.

Along with VMware’s products, XenServer has the proven track record and presence in the business sector, but its openness and excellent Linux support has also made it attractive in the scientific and academic environments.

4 Evaluation Criteria

Evaluating complex products such as cloud platforms requires a common set of criteria for comparison [2], [3]. The primary goal for our criteria was to establish a comparison baseline for open-source IaaS products, which we later extended to include commercial/closed products. Some criteria, especially high-level ones, can be applied to other cloud models (PaaS and SaaS). The criteria are divided over six main groups: storage, virtualization, network, management, security and vendor support. The criteria admittedly favor platforms adapted to Linux workloads and hosts, and functionalities over usability and integration, which can be corrected using weights.

We have graded individual criteria in the range of 0 to 3, where 0 designates no support and 3 designates full support. In cases where support for a feature relies on the underlying operating system drivers or tools, we assigned the grade of 2 if the operating system support is mature. We describe each group in short, as their full description is provided in our earlier work and is out of scope of this paper.

Storage Criteria. Management and functionalities of virtual machine storage are critical to achieve flexible and scalable implementations. Storage-related criteria focus on main technologies which implement cloud storage: direct-attached storage (DAS), storage area networks (SAN), and network-attached storage (NAS), as well as backup features and technologies. Main technology groups are further divided into specific technologies, e.g. file system and replication support.

Virtualization Criteria. Virtualization technologies are the enablers of cloud computing. Compared to commercial products, open-source platforms support a wider range of virtualization technologies, often three or more open-source and commercial hypervisors. Virtualization criteria focus features such as VM type and technology, quotas, migration and cloning, resource prioritization, hot configuration and provisioning. Platforms supporting a wide range of hypervisors fare better than platforms that focus on a single hypervisor, regardless of the level of supported functionalities, which we tried to amend by grading the level of support.

Network Criteria. Network connectivity binds the cloud components together, and connects it with its administrators and end-users. Network support is covered with criteria describing support for VLAN services such as tagging (IEEE 802.1q), network management and isolation, firewall support via network filtering, Ethernet quality of service (IEEE 802.1p), and integration capabilities such as IPv6 and virtual private networks for access and management.

Management Criteria. Comprehensive and usable management is directly related to the way a virtual machine cloud is handled. Criteria describe management facilities of a platform and focus on host-guest OS integration, individual and mass management of both hosts and guests, exposed and consumed APIs, data collection for billing and reporting, automatic recovery, high availability and alerting features.

Security Criteria. Cloud installations can pose a significant security risk. We describe security-related capabilities of a platform such as data encryption, directory services integration, authorization levels and auditing events for specific resources (e.g. VM server or storage servers). Security report types available to cloud administrators, third-party product evaluations and certifications of compliance with security standards, and secure management access are also looked into.

Community and Vendor Support Criteria. Product support is often divided between a user community and a vendor. Community-related criteria deal with freely available support channels of the user or developer communities gathered around the product, and their quality. Vendor-related criteria concern with direct vendor support channels and the possibility of a SLA contract. We also cover methods of customer relations such as public issue tracking, proactive updates and CRM-like approaches, completeness of open-source or free product versions and related documentation, track record, future viability, and the possibility of third-party auditing.

4.1 Feature Comparison

We have compared the features of Eucalyptus Community Cloud 2.0.3 (ECC), OpenNebula 3.2 (OpenNebula), VMware vCloud 5 (vCloud), Microsoft Private Cloud 2012 (MSPC), and Citrix CloudStack 2.2 (CloudStack). Table 1 presents a sum of grades by criteria group, normalized to the maximum possible score per group. The full evaluation table contains 97 grades per product, 485 in total.

Table 1. Feature comparison with evaluation criteria

Criteria group	ECC	OpenNebula	vCloud	MSPC	CloudStack
Storage	22 (49%)	24 (53%)	19 (42%)	21 (47%)	26 (58%)
Virtualization	18 (27%)	30 (45%)	54 (82%)	21 (32%)	51 (77%)
Management	33 (48%)	44 (64%)	58 (84%)	37 (54%)	45 (65%)
Network	11 (41%)	6 (22%)	14 (52%)	10 (37%)	14 (52%)
Security	13 (39%)	16 (48%)	26 (79%)	14 (42%)	14 (42%)
Vendor Support	30 (59%)	37 (73%)	39 (76%)	23 (45%)	37 (73%)
Total score	**127 (44%)**	**157 (54%)**	**210 (72%)**	**126 (43%)**	**187 (64%)**

As expected, vCloud is the most feature-complete solution, with CloudStack and OpenNebula following suit. CloudStack has a capable hypervisor, while Microsoft has the edge with its Windows ecosystem hardware support, management and integration, as well as vendor support, but most of the advanced features it offers apply only for Windows workloads. Open-source products exhibit a lower level of component integration, and excel in some areas while lacking in other.

5 Hypervisor Performance Comparison of Commercial/Closed and Open-Source Cloud Solutions

While virtualization is not a necessary part of cloud computing solutions, it is almost universally present for the convenience it brings to the implementation of elasticity and automation [4]. Virtualization is commonly achieved by emulating complete virtual machines under the control of hypervisor software (as opposed to more light-weight virtualization environments [5]), making such software essential for stability, security and performance of the whole cloud infrastructure. The scientific computing environment is somewhat special as it commonly uses private or semi-private infrastructure (i.e. where both the users and the infrastructure are known and controlled by a single entity), which makes both security and software stability less risky. System performance of virtualized environments is of importance to many potential users, which drove us to investigate this aspect of cloud computing. We have set our goal to evaluate common computing workloads and provide a comprehensive answer on the influence of modern hypervisor software on the overall system performance.

Our evaluation was centered on four of the most common virtualization products: VMware ESXi 5.0, Citrix XenServer 6.0, Microsoft Hyper-V 2008 R2 SP1, and KVM from Linux 2.6.32. The baseline was an un-virtualized system on an identical hardware platform (2x Xeon 5460 CPUs, 16 GB of memory, 2x 15 kRPM SAS disk drives). We have used the CentOS 6.0 operating system for the virtual machines, with the same virtual machine disk image used on all hypervisors. The latter was necessary to reduce the number of variables and outside influences on the measured system, but has lead us to conclude that, sadly, there is no standard for migrating virtual machines between virtualization environments adopted by all the major vendors and projects. We hoped that the Open Virtualization Format [6] ratified as an ANSI standard would have been widely adopted, but its support is notably absent from Microsoft's products.

5.1 Notes on the Hypervisors

Of the evaluated hypervisors (summarized in Table 2), ESXi, XenServer and Hyper-V can be described as “Type 1” hypervisor [7], running on “bare-metal” without a supporting operating system, while KVM is decidedly “Type 2“, making full use of regular Linux kernel features and subsystems in its operation. The most important assumption which can be derived from this distinction is that the Type 1 hypervisors are more light-weight and introduce less latency. One of our goals was to verify the validity of this assumption for the chosen hypervisors and the guest operating system.

Another goal was to check the impact of paravirtualization support for certain hypervisors, most notably Xen (used in XenServer) and KVM. Xen's paravirtualization capabilities are much more extensive than KVM's, which led us to expect significant performance improvements. Neither VMware not KVM provide paravirtualization at the level of virtual memory management or other operating system components which would influence our benchmark.

Table 2. Hypervisors tested in this work

Hypervisor	License	Type	Cloud product
VMware ESXi 5.0	Commercial	Type 1	vCloud
Citrix XenServer 6.0	Commercial/GPL	Type 1	CloudStack
Microsoft Hyper-V 2008 R2 SP1	Commercial	Type 1	Microsoft Private Cloud
CentOS 6.0 KVM (Linux 2.6.32)	GPL	Type 2	OpenNebula, Eucalyptus...

According to industry surveys [8], [9], the three commercial virtualization solutions dominate the market, with a cumulative share upwards of 80% in product deployments. Of these, VMware's products are more popular by a significant margin. Scientific and academic environments are traditionally very strongly tied to open source solutions for reasons of cost-efficiency and historically better support for specific workloads. We have observed ever stronger industry lobbying emphasizing “enterprise-grade” features and more refined management tools, which could mean a wider adoption of commercial hypervisors in scientific computing environments.

5.2 Simulating Scientific Workloads

There is of course no single definition of a "scientific workload", but there are regular attempts to characterize typical workloads of certain types of scientific environments [10-13]. We decided on an approach of describing typical performance by using a number of benchmark suites, with the intent that the overall results show the relative differences between the tested systems, with respect to CPU, memory, and OS interaction performance. Table 3 summarizes the selected benchmark suites.

Table 3. Benchmark suites used in this work

Benchmark	Platform/environment	Threading	Result interpretation
DaCapo 9.12	Java (OpenJDK 6 b17)	Mixed	Time (lower is better)
Pysysbench 0.9	Python/C (Python 2.6)	Multi-process	Throughput (higher is better)
Bullet Cache 1.0.1	C, C++ (gcc 4.4.6)	Multi-threaded	Transactions per second (higher is better)
SciMark2/C	C (gcc 4.4.6)	Single-threaded	Throughput (higher is better)

The DaCapo benchmark suite [11] evaluates real-world performance by running a set of open source applications of a varied behavior. From this set of benchmarks we have used the following: avrora (AVR microcontroller simulation, single-threaded), batik (Scalable Vector Graphics rendering, single-threaded), eclipse (Eclipse development environment, multi-threaded), h2 (in-memory database, multi-threaded), jython (Python interpreter in Java, multi-threaded), luindex (Lucene full-text searching library, single-threaded), lusearch (Lucene, multi-threaded), pmd (source code analyzer, multi-threaded), sunflow (ray-tracing graphics, multi-threaded), and xalan (XML transformer, multi-threaded). The DaCapo results are reported as average execution time for each benchmark, in milliseconds.

The Pysysbench benchmark measures basic throughput of common low-level CPU-intensive tasks implemented in C libraries called from Python: Hash-SHA256, Hash-SHA512, Zlib-Compress, Zlib-Decompress, Socket-Syscalls (a loop executing the socket() and close() system calls), Socket-OnePipe (a loop transferring 1 byte of data over a pair of sockets). All tasks are executed in parallel processes and the results indicate throughput (amount of data processed or the number of operations per second), as a composite dimensionless score.

The Bullet Cache benchmark runs the multi-threaded Bullet Cache[2] [12] memory database with 5 threads on the server and on the client side, with a varied number of client connections, and reports the number of transactions per second, in thousands.

The SciMark2/C benchmark is the C implementation of the SciMark2 benchmark, implementing a set of scientific numerical algorithms: FFT, SOR, Monte Carlo integration, sparse matrix multiplication, and dense LU matrix factorization. The

[2] Bullet Cache / MultiDomain Cache Daemon, `http://mdcached.sourceforge.net/`

benchmark was used in the "large" configuration and the reported results present an approximate MFLOPS performance.

All benchmarks were installed and ran from a single virtual machine image, converted as necessary for various hypervisors. The hypervisors themselves were installed on identical hardware systems equipped with two Xeon 5460 CPUs and 16 GB of memory. The virtual machines were configured with 8 virtual CPUs and 8 GB of memory, except in the case of Hyper-V where, due to its limitations, the virtual machine was configured with 4 virtual CPUs.

5.3 Benchmark Results

The chosen benchmark suites produce detailed output for each individual benchmark, which can be used to interpret specific aspects of the tested systems (discussed in following sections). The benchmark results are the averages of five consecutive runs, and the overall results are presented summarized per benchmark and per hypervisor in Table 4 and Figure 1.

Table 4. Hypervisor benchmark results (compared to non-virtualized baseline)

	ESXi	XenServer	Hyper-V	KVM	Non-virt.
DaCapo	5530 (+25%)	6010 (+37%)	6240 (+42%)	6020 (+37%)	4390
Pysysbench	1013 (-5%)	1043 (-2%)	532 (-50%)	1020 (-4%)	1068
BulletCache	217 (-46%)	218 (-46%)	165 (-59%)	184 (-54%)	401
SciMark2/C	522 (-2%)	520 (-3%)	507 (-5%)	511 (-4%)	533

The results show that the performance of current generation of hypervisors is reasonable and there are, at first sight, no significant outstanding problems. The one exception is Microsoft's Hyper-V whose results are skewed in aggressively multi-threaded benchmarks (Pysysmark and Bullet Cache) because it does not support more than 4 virtual CPUs. We draw the following conclusions from the results:

1. Performance differences among the latest versions of the hypervisors, tested under the specified operating system(s) and hardware, are mostly within 10%. The performance difference to the non-virtualized case, however, can be larger than 50% (in the DaCapo and Bullet Cache benchmarks);
2. Single-threaded, purely computational workloads (SciMark2/C) exhibit the smallest performance impact, between 2% (ESXi and XenServer) and 5% (Hyper-V);
3. The largest differences are in extensively multithreaded workloads (Bullet Cache);
4. From the detailed results we can also conclude that simple system calls do not exhibit a significant performance impact. Combined with the previous point it is likely that execution context switching comes with huge penalties in virtualized environments. However, more testing is required on newer equipment (CPUs implementing Nested Page Tables) to verify that the impact is still significant.

The KVM hypervisor exhibited good overall performance, but it was never at the top. The smallest virtualization impact (4%) was measured for SciMark2 and Pysysbench but this result is still far from the best (2%, measured for the XenServer).

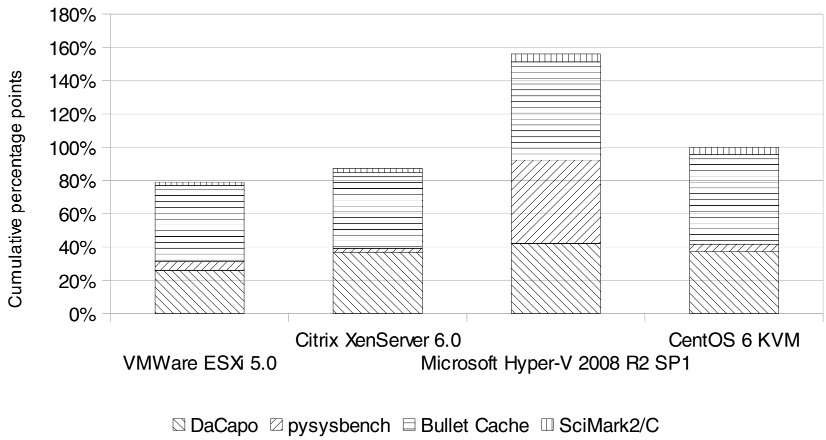


Fig. 1. Relative performance impact of virtualized environments, compared to non-virtual environments, by cumulative percentage points

The absolute performance impact of KVM compared to the non-virtualized case is large (figure 1). The relative impact when compared to the leading hypervisor (ESXi) is much smaller, 15% in the worst case (Bullet Cache) but only 0.7% in the best case (Pysysbench). The difference for the single-threaded compute-intensive workload (SciMark2) is 2% and nearly 9% for the mixed application workload (DaCapo). Whether this is a problem in practice depends on a specific deployment, but we are confident that the performance is suitable for most uses. XenServer and Xen hypervisors share the same code base and we expect them to have identical performance.

6 Conclusion and Future Work

We have given a brief overview of several open-source and commercial/closed private-cloud platforms. Present popularity of cloud computing paradigm and headroom for further development guarantee continuous growth and improvement of cloud platforms. Using a set of criteria developed for evaluating open-source platforms, we have evaluated the features of leading commercial/closed solutions, and confirmed that open-source and commercial solutions have comparable features.

We also compared performance of hypervisors running at the core of evaluated cloud products. In an attempt to simulate scientific workloads, we used a set of benchmark suites testing specific system aspects – CPU, memory, and operating system interaction performance. Benchmark results led us to conclude that modern hypervisors performance is close to non-virtualized case with single-threaded workloads. Large performance differences were observed with multi-threaded workloads, both between the hypervisors and compared to non-virtualized hardware. The tests were made on an older generation of hardware, and we plan to continue the tests once we have state-of-the-art equipment available.

References

1. Iosup, A., Ostermann, S., Yigitbasi, N., et al.: Performance Analysis of Cloud Computing Ser-vices for Many-Tasks Scientific Computing. IEEE Transactions on Parallel and Distributed Systems 22, 931–945 (2011)
2. Voras, I., Mihaljević, B., Orlić, M., et al.: Evaluating Open-Source Cloud Computing Solutions. In: Proc. of the 34th International Convention MIPRO 2011, Zagreb, pp. 209–214 (2011).
3. Voras, I., Mihaljević, B., Orlić, M.: Criteria for Evaluation of Open Source Cloud Computing Solutions. In: Proc. of the ITI 2011, 33rd International Conference on Information Technology Interfaces (ITI), Dubrovnik, Croatia, pp. 137–142 (2011)
4. Armbrust, B., Griffith, R., Joseph, A.D., et al.: A view of cloud computing. Communications of the ACM 53, 50–58 (2010)
5. Kamp, P.-H., Watson, R.N.M.: Jails: Confining the Omnipotent Root. In: Proc. of the 2nd International System Administration and Networking Conference, SANE 2000 (2000)
6. VMware Inc.: The Open Virtual Machine Format Whitepaper for OVF Specification, version 0.9, `http://www.vmware.com/pdf/ovf_whitepaper_specification.pdf`
7. Goldberg, R.P.: Architectural Principles for Virtual Computer Systems. Doctoral dissertation. Harvard University (1972)
8. Bigelow, S.J.: State of virtualization and cloud computing (2011), `http://searchdatacenter.techtarget.com/feature/State-of-virtualization-and-cloud-computing-2011`
9. Citrix Systems, Inc.: Cloud.com 2011 Cloud Computing Outlook Survey Results (2011), `http://www.cloud.com/cloud-computing-outlook/survey.pdf`
10. Bradley, D., Cybenko II, G., Larson, J., et al.: Supercomputer workload decomposition and analysis. In: Proc. of the 5th International Conference on Supercomputing, pp. 458–467. ACM (1991)
11. Cirne, W., Berman, F.: A comprehensive model of the super-computer workload. In: Proc. of the Fourth Annual IEEE International Workshop on Workload Characterization WWC4 Cat No01EX538, pp. 140–148 (2001)
12. Numrich, R.: A note on scaling the Linpack benchmark. Journal of Parallel and Distributed Computing 67, 491–498 (2007)
13. Gray, L., Kumar, A., Li, H.: Workload Characterization of the SPECpower_ssj2008 Benchmark. In: Kounev, S., Gorton, I., Sachs, K. (eds.) Performance Evaluation Metrics Models and Benchmarks, pp. 262–282. Springer, Heidelberg (2008)
14. Blackburn, S.M., Guyer, S.Z., Hirzel, M., et al.: The DaCapo benchmarks. In: Object Oriented Programming Systems Languages and Applications OOPSLA, vol. 41, p. 169 (2006)
15. Voras, I.: Cache server for distributed applications adapted to multicore systems. Doctoral dissertation, Faculty of Electrical Engineering and Computing, University of Zagreb (2011)

A Self-adaptive Multi-Agent System for Abnormal Behavior Detection in Maritime Surveillance

Nicolas Brax, Eric Andonoff, and Marie-Pierre Gleizes

IRIT, 118 Rte de Narbonne, 31062 Toulouse Cedex, France
{nicolas.brax,eric.andonoff,marie-pierre.gleizes}@irit.fr

Abstract. This paper presents a Multi-Agent System (MAS) dedicated to abnormal behaviors detection and alerts triggering in the maritime surveillance area. This MAS uses anomalies issued from an experienced Rule Engine implementing maritime regulation. It evaluates ships behavior cumulating the importance of related anomalies and triggers relevant alerts towards human operators involved in maritime surveillance. These human operators evaluate triggered alerts and confirm or invalidate them. Invalidated alerts are sent back to the MAS for a learning step since it self-adapts anomalies values to be consistent with human operators feedbacks. This MAS is implemented in the context of the project I2C, an EU funded project[1] dedicated to abnormal ships behavior detection and early identification of threats such as oil slick, illegal fishing, or lucrative criminal activities (e.g. goods, drugs, or weapons smuggling).

Keywords: Maritime Surveillance, Alert, Learning, Adaptive MAS.

1 Introduction

Nowadays, maritime activity, which has grown rapidly in recent years, provides support for numerous and various traffic (e.g. arms dealing, goods or drug smuggling) or illicit activities (e.g. illegal fishing). As a consequence, States having long coastlines or vulnerable trading lanes want to have efficient maritime surveillance systems helping them in identifying these criminal activities in order to deal with the threats they represent [1].

To be efficient, a maritime surveillance system must be able to permanently track and monitor all type of maritime activities in order to detect abnormal behaviors, to analyse them and to early identify threatening situations. However, currently there does not exist a system that can handle all the necessary information needed to detect all abnormal behaviours. More precisely such a system must provide [2]:

[1] Integrated system for interoperable sensors and Information sources for common abnormal vessel behavior detection and Collaborative identification of threat. See www.i2c.eu for more information.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 174–185, 2012.

- a permanent and all weather coverage of border maritime areas,
- a continuous collection and fusion of heterogeneous data provided by various types of sensors deployed on coast and on mobile platforms and other information from external sources,
- an automatic detection of abnormal ship behaviors (in track and performed activity) and the triggering of alerts if these abnormal behaviors represent potential threats,
- the understanding of these threats in order to allow decisional authorities to deal with them,
- an adapted interface intended to human operators involved in maritime surveillance.

Significant technical progresses have been made in wide maritime area coverage by different sets of sensors [3], in heterogeneous data processing and fusion [4], and in detection of abnormal behaviors methodologies [5] that could be usefully integrated together to built up a new generation of maritime surveillance systems for efficient security applications in high density traffics area [6].

The aim of I2C is to provide maritime surveillance actors with such a system [7]. However, this paper does not report propositions to track ship behavior or to integrate collected information into I2C. It rather focuses on automatic detection of abnormal ship behavior and alert triggering, which is a challenging issue in surveillance systems. The system interacts with and helps human operators in identifying suspicious behaviors among the large amount of ships they observe (hundreds of ships) and for which it is necessary to trigger an alert.

No matter of the application area (maritime surveillance or medical [8], sociological [9], network security areas [10]), we distinguish between two kinds of systems. First, those that model authorised behaviors, using neural networks for instance, and indicate that the observed behavior is abnormal if not described in the model [11, 12]. However, in such systems, unknown behavior is considered as abnormal, which is not necessarily true in our context. Secondly, there are systems modelling domain regulations (*e.g.* maritime regulations) using rules and rule engines to identify abnormal behaviors according to observed events in the supervised areas [13–16]. This solution detects anomalies corresponding to unexpected ship behaviors. But it is inadequate and has to be improved in the maritime surveillance context for three main reasons. First, most of the time, abnormal ship behavior is the result of the violation of several rules simultaneously, and each violation considered independently of one another is not an anomaly and does not require alert triggering. Unfortunately, provided rule engines do not deal with that. Second, maritime regulation is very complex and moreover, alerts are sometimes ship-dependant. Therefore, it is difficult to model this dependence using rules. Third, by using a fully rule-based approach it is impossible to discover new behavior of law offenders who are creative, financially and politically motivated. The system has to be able to identify new abnormal behaviors, and ideally it also has to be able to learn the new strategies by the law offenders [17].

In the project I2C, we advocate to combine a Rule Engine (RE) implementing maritime regulations and an adaptive Multi-Agent System (MAS) [18–20] responsible for ship behavior evaluation and alert triggering. The MAS evaluates the ships behaviours according to the anomalies identified by the RE and triggers alerts. In order to be able to compute this evaluation, the MAS models anomalies and their values as agents and features learning capacity in order to self-adapt anomaly values according to feedbacks of human operators involved in maritime surveillance. This MAS is an important added-value in I2C since maritime surveillance systems have to face evolving situations and law offender behaviors are often unpredictable and have to be discovered [21]. The learning capacity of this MAS is an important skill of I2C [22], and the major contribution of the paper is the proposed self-adaptive MAS for abnormal behavior detection.

The outline of the paper is as follows. Section 2 introduces the architecture of the I2C system and focuses on the role of the MAS. Section 3 describes the Operative Multi-Agent System (OpMAS), the first MAS component dedicated to the evaluation of ships behavior and alert triggering. Section 4 presents the Parameter Adjustment Multi-Agent System (PAMAS), the second MAS component introduced for anomalies values modelling and learning. Section 5 illustrates alerts triggering through an example. Finally, section 6 concludes the paper.

2 Architecture of I2C Maritime Surveillance System

This section first introduces the general architecture of I2C and then focuses on the role of the MAS it integrates.

2.1 General Architecture of I2C

The architecture of I2C is presented in Figure 1. It integrates several components which are sensors and databases, an intelligent traffic picture called SITU, a RE

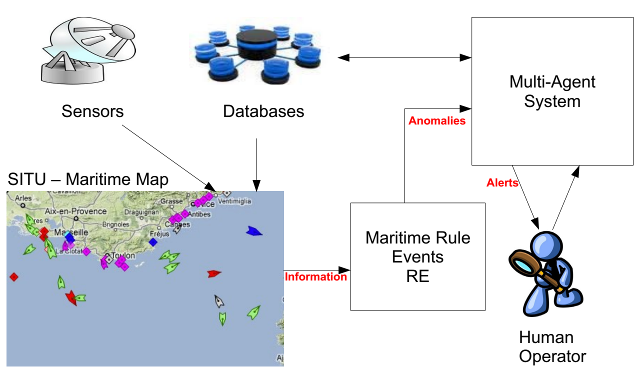


Fig. 1. I2C architecture

dedicated to anomalies detection from maritime regulations and observed events in the supervised area (e.g. ship stopped in international waters), and a MAS responsible for alert triggering if the suspicion level of the cumulated anomalies is too high (*e.g.* illegal fishing, smuggling of drugs). These alerts are then evaluated by human operators involved in maritime surveillance. If they correspond to threatening situations, authorities perform actions in order to deal with them (e.g. boarding a ship after triggering a drug smuggling alert).

More precisely, SITU integrates and merges information issued from different sources which are both sensors and databases [23] in order to provide a maritime map representing positions of ships in real time. Information from SITU (e.g. position of a ship, speed of a ship, black-listed ship) are then sent to the RE whose aim is to detect anomalies which correspond to suspicious events. The RE integrates maritime regulations described as atomic rules whose conditions refer to information issued from SITU. For instance the following rule expresses that a speed anomaly is detected during an excessive speed of a ship in an harbour: *IF speed>15 AND area="Harbour" THEN Detect(Speed_anomaly).*

These anomalies are then sent to the MAS whose aim is to trigger relevant alerts towards human operators involved in maritime surveillance. The human operators validate or invalidate these triggered alerts. Validated alerts are transferred to a group of experts for identification of threats and eventual actions (*e.g.* boarding of a suspicious ship). Invalidated alerts are sent back to the MAS for learning.

2.2 Role of the Multi-Agent System

As described in the introduction, the RE that handles maritime regulations is combined with our adaptive MAS for abnormal behavior detection and alert triggering. The role of the MAS is threefold. First, the MAS evaluates ship behavior as a numeric value, called SBv, measuring the importance of anomalies identified by the RE. Second, it triggers alerts towards human operators if this numeric value is greater than a threshold given by the human operators. Third, it learns by self-adaptation the importance of anomalies when the human operators invalidate triggered alerts.

3 Operative MAS to Compute Ship Behaviour

This section presents our MAS architecture and then focuses on the MAS component dedicated to ship behavior evaluation and alert triggering.

3.1 MAS Architecture

As illustrated in Figure 2, our MAS integrates two main components: OpMAS and PAMAS. In OpMAS, a ship-agent is created for each ship monitored by I2C in the real world. OpMAS evaluates ships behaviors considering issued anomalies from the RE. It computes for each ship-agent its corresponding SBv, i.e. a numeric value representing the ship behavior cumulating the importance of its anomalies. It triggers an alert if this value is greater than a threshold defined by a human operator. On the other side, PAMAS models the importance (value) of anomalies through anomaly-agents and parameters-agents. PAMAS distinguishes three values for each anomaly, called parameters, which correspond to the value the anomaly has when it appears (***Init***), the increasing value it has while it lasts (***Incr***) and the decreasing value it has after its disappearance (***Decr)***.

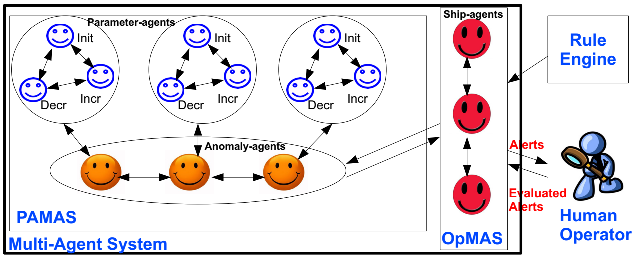


Fig. 2. OpMAS and PAMAS Components

3.2 Ship-Agents in OpMAS

Basically, each ship-agent represents a ship and it is filled with skills and abilities in order to reach its goals. First, it integrates abilities to interact with three other components of I2C:

- the RE in order to receive detected anomalies which will be used to evaluate ships behavior according to their corresponding values described in PAMAS,
- PAMAS in order to ask for values of anomalies or for a learning step by self-adaptation of anomalies values,
- human operators who receive triggered alerts from OpMAS, which after evaluation and if invalidated are sent back to OpMAS for a learning step.

Second, a ship-agent associates to the ship it represents a behavior (suspicion) value, called SBv. The higher the value, the more the ship behavior is suspicious. This value is computed taking into account anomalies sent by the RE and according to the values of corresponding parameters as estimated by PAMAS. The following function is used to compute this value:

$$SBv = \sum_{i=1}^{n} Init(a_i) * Nb('Init', a_i) + Incr(a_i) * Nb('Incr', a_i) - Decr(a_i) * Nb('Decr', a_i)$$

The idea is to cumulate the importance of the different anomalies of a ship according to the values of their parameters and taking into account how long these anomalies last. As illustrated in Figure 3, a ship-agent also gives a graphical representation of its behavior as a curve and it also gives the corresponding inequality considering the threshold fixed by human operators. For instance, Figure 3 illustrates a ship behavior involving anomaly a_1 from step0 to step13, and anomalies a_1 and a_2 from step14 to step19. We have the following value for

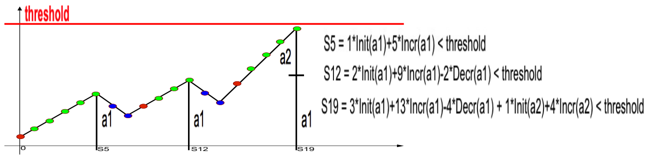


Fig. 3. Computing ship behavior

SBv at step5: $1*Init(a_1)+5*Incr(a_1)$. Indeed, at step5 a_1 appeared 1 time and lasted 5 times. At step 12, a_1 appeared 2 times, lasted 9 times and decreased 2 times. At step19, both a_1 and a_2 are combined; a_1 appeared 3 times, lasted 13 times, decreased 4 times while a_2 appeared 1 time and lasted 4 times. For each of these values, the ship-agent can deduce the corresponding inequality expressing that the value is lower than the threshold.

Third, a ship-agent interacts with other ship-agents individually using direct messages or collectively using indirect messages. Regarding collective behavior, some illicit ship behaviors cases are difficult to identify as they result from a collective actions from supervised ships, each of them acting normally. Indeed, as they act as normal, the RE does not identify any anomaly since they do not individually contravene maritime regulations. To deal with these particular cases, OpMAS introduces stigmergy based detection mechanisms. As a consequence, a ship-agent can mark its location in the environment in order to inform the other ship-agents of an event that can be relevant to them. Just as pheromone tracks of ants, these marks are cheap in resources and can be a useful source of information [24]. A mark is deposed by a ship-agent with a critical value and a diffusion value giving the lifetime of the mark. When another ship-agent perceives a mark in its environment, it can in turn consider it as a relevant anomaly and then uses it to update its behavior suspicion value.

Finally, regarding individual interactions, a ship-agent can ask for information or data about another ship-agent, for example its suspicious level. A ship-agent can also send an alert to another ship-agent in order to possibly propagate it. Besides, it can send direct messages to another known ship-agent, either because it is in its direct neighbourhood or because of a mark left by the agent. This cooperation is used by the agents to determine the ones that are involved in a given anomaly and, accordingly, the ones that have to be marked as an alert.

4 Parameter MAS to Self-adjust Parameters

This section presents PAMAS that tunes values of anomalies involved in alerts according to human operator feedbacks. It first introduces PAMAS agents, then explains how this tuning is performed, using a self-adjusting approach, and finally how situation historic can be taken into account to improve the tuning.

4.1 PAMAS Agents

To each type of anomaly sent to OpMAS by the RE, there is a corresponding agent in PAMAS, called anomaly-agent. This anomaly-agent manages the value of the anomaly as a function of three parameters, each represented as a parameter-agent:

– the *initial value*, *Init*, of the anomaly which can be seen as the prime importance of the anomaly,
– the *increment value*, *Incr*, which is the increasing importance of the anomaly while it lasts and,
– the *decrement value*, *Decr*, which represents the time the anomaly is kept in memory once it has disappeared.

As these parameters are a representation of the human operators knowledge, it is difficult to give them a value. Therefore, PAMAS provides initial values to these parameters and then tunes them according to the human operator feedbacks in order to reach acceptable values which correspond to a stable state of the MAS, and consequently to trigger relevant alerts. This self-tuning is possible due to the cooperation between anomaly-agents and parameter-agents.

4.2 Tuning Parameters

Whenever a ship-agent retrieves the values of some anomalies and combines them to compute SBv, PAMAS expects a feedback on these values. This feedback can be positive if OpMAS raises an alert and the operator confirms it or, if there is no confirmation, the MAS assumes the values are good enough. However, the feedback can also be negative if OpMAS raises an alert and the operator negates it, or when the system did not raise an alert and the operator spots one. In this case, PAMAS needs to tune the parameters values of the corresponding anomalies in order to correct the mistake it did. Basically, the anomalies-agents must lower or increase the values of anomalies according to the human operator feedbacks and the defined threshold.

In order to do this, anomaly-agents use two mechanisms to tune themselves. On one hand, a critical level is associated to each anomaly-agent. This critical level is calculated depending on the importance of the anomaly in the alert, and also on the use of this anomaly by other ship-agents. The suspicious value of a ship is directly influenced by the anomalies in proportion to their importance. And the constraints on an anomaly are dependent of its frequency in the behaviors of the other ships. Therefore, we can say that the critical level will weight the adjustment of the anomaly in a proportional way.

On the other hand, in most cases, an alert is the combination of several anomalies. Therefore, the different anomaly-agents involved have to cooperate in order to tune their values and adjust the global value of the ship behavior according to human operators feedbacks. This cooperation is based on the exchange of the critical level values of the anomalies. Knowing the critical level of the other involved anomalies, the agents can build a rank system and then update their critical level according to the other agents.

We can actually say that the basic adjustment of the anomaly-agents is based on the critical level of each one, which will weight the tuning of the parameters. Such a method, however, have a negative impact on the system. Indeed, when an anomaly-agent tunes itself, it can reach a state where its value goes against a past situation with no negative feedback. Then, if this situation happens again, PAMAS will have to tune the parameter values again. This can lead to a system that will never converge into a stable state and so will never be able to raise relevant alerts towards human operators. As a consequence, we have improved the learning of the MAS as explained in the next section.

4.3 Collective Learning from Past Situations

We propose to take into account past situations while PAMAS tunes itself. These situations can be of two kinds:

- situations where the system is in a highly constraint state, meaning that the value of at least one of the anomalies is estimated to be correct by the system because a change of this value would change the "alert state" of the corresponding ship,
- situations which are known to be a negative feedback in order to avoid the repetition of these negative feedbacks.

These situations are used to tune the values of anomaly-agents parameters. Each anomaly-agent tunes itself according to its own critical level and the critical levels of its neighbourhood, i.e. the other involved anomaly-agents. Basically, the agents learn from their history to avoid the repetition of past errors and this ensures the convergence of the system into a stable state.

In order to illustrate this idea, we take the example represented in Figure 3 and we suppose that a human operator sends a negative feedback at step19 indicating that SBv should be greater than the threshold. As explained before, this negative feedback induces a learning step from PAMAS and, more precisely, a self-adaptation of the parameters of the anomalies a_1 and a_2, since these two anomalies are involved in the considered ship. In order to improve the tuning of these parameters, PAMAS considers the situation of step12 in addition to the situation of step19 and it self-adjusts the values of a_1 and a_2 considering the two following inequalities:

$$S12 : 2 * Init(a_1) + 9 * Incr(a_1) - 2 * Decr(a_1) < threshold$$
$$S19 : 3 * Init(a_1) + 13 * Incr(a_1) - 4 * Decr(a_1) + 1 * Init(a_2) + 4 * Incr(a_2) < threshold$$

Indeed, PAMAS considers the situation of step19 as it is a negative feedback from the human operator and the inequality used for self-adaptation of parameters values is $3*Init(a_1)+13*Incr(a_1)-4*Decr(a_1)+1*Init(a_2)+4*Incr(a_2) > threshold$ as the MAS did a mistake when calculating SBv which must be greater than the given threshold. It also considers step12 since the system is in a highly constraint state. Thus, in order to deal with the situation of step19 (which is incorrect), PAMAS will increase the value of a1, without contravening the situation of step12 (which is correct).

5 Alert Triggering Example

This section illustrates the behaviors of the OpMAS and PAMAS agents through an example. This example considers three fishing ships A, B and C, and an unknown ship out of the surveillance coverage. This example also considers only the "stop in open sea" anomaly. This anomaly alone does not lead to an alert, but we will see how the collective behavior of the agents can trigger a relevant alert.

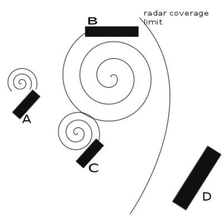


Fig. 4. Fishing ships A, B and C; unknown ship D

5.1 Individual Behavior in PAMAS

According to Figure 5.a, at *step0*, the fishing ship A stops in open sea. This event is detected by the RE which sends an anomaly to OpMAS. The corresponding ship-agent retrieves the value of the anomaly in PAMAS and computes SBv, the value of the behavior of the ship. The current anomaly is not enough to raise an alert, but the suspicious level of the ship is increased and the ship-agent leaves a mark in the environment, reporting the anomaly for others.

At *step10*, we can see on Figure 5.b that the fishing ship B also stops in open sea, near where the first mark was left by the ship A. The RE sends another anomaly to OpMAS in order to compute the value of the ship behavior. But this time, the ship-agent uses the mark in the environment in addition of the value

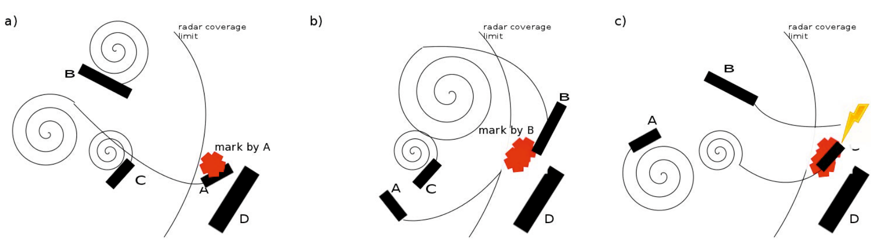


Fig. 5. a) *Step0*; b) *Step10*; c) *Step14*

sent by PAMAS to increase the suspicious level. Again, this is not enough to raise an alert but another mark is left, strengthening the first one.

At *step14*, the third fishing ship stops near the previous locations. This time, the value computed by OpMAS uses the value received from PAMAS as well as the two marks previously left by the other ship-agents. This computation leads to an increase of the suspicious level of the third ship, which is sufficient to raise an alert on this ship.

5.2 Collective Behavior in OpMAS

We have just seen that the ship-agents in OpMAS use the anomalies sent by the RE as well as the marks left in the environment in order to compute the suspicious level of the ships they represent. Furthermore, the ship-agent C which is detected as suspicious can use the information on the marks in the environment. Indeed, we can consider that if several ships have stopped at the same location in a limited amount of time, and one has a suspicious level high enough to raise an alert, the other ships are suspicious too. As shown in Figure 6, the ship-agent C will contact the two other fishing ship-agents and systematically inform them about the alert. Depending on the other available informations, the MAS can even detail the alert. In our example, the three involved ships are fishing ships. The MAS can consequently assume that the three detected stops are part of a transshipment operation and then spot the possible presence of a refrigerated ship, here our unknown ship D. Finally, the alert triggered by the MAS is *Illegal fishing*.

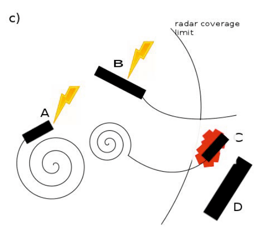


Fig. 6. The alert is spread to other ships

6 Conclusion

In this paper, we have presented a Multi-Agent System (MAS) dedicated to abnormal behavior detection and alert raising in the maritime surveillance domain. This work takes place in the European funded project I2C. Based on anomalies sent by an experienced Rule Engine (RE), the MAS evaluates ships behavior according to the importance (value) of the related anomalies and raises relevant alerts towards human operators involved in maritime surveillance. One of the main aspect of the MAS is its ability to learn from human operators feedbacks and it self-adapts the values of the anomalies to be consistent with these feedbacks. More precisely, the paper has presented the two main components of the MAS. First, the Operative Multi-Agent System (OpMAS) which represents the real ships as agents, computes ship behavior values and sends alerts to human operators. Second, the Parameter Adjustment Multi-Agent System (PAMAS) is used to self-adapt the values of the anomalies according to the feedbacks from the human operators and takes into account an historic of events through past situations. This tuning allows the whole system to be consistent with the feedbacks and improves the relevance of the alerts raised by OpMAS.

This MAS is implemented in a simulation platform which integrates several other components in order to test and use our proposition. One component simulates the rule engine and sends anomalies to the OpMAS. Another simulates the human operators, receiving the alerts from the OpMAS and sending the needed feedbacks. Finally, there is also a component building random intelligent scenarios for each ship and managing the whole platform to maintain its relevance. Our current preoccupation is the implementation of the tuning of the PAMAS agents behaviour.

The next step will be the evaluation of the efficiency and the effectiveness of the MAS with respect to the real-world data provided by I2C, *i.e.* with a natural number of ships and a relevant number of anomalies.

Acknowledgement. The authors would like to thank DCNS who leads the European project I2C and who is involved with us and Clear Priority in the work packages dedicated to abnormal behavior detection. The authors would also like to thank Henric Johnson, from the Blekinge Tekniska Högskola in Karlskrona – Sweden, for the fruitful discussions we had together on this topic and for his comments on earlier versions of the paper.

References

1. Henocque, Y., Lafon, X.: EUs Strategy on Maritime & Environmental Issues in the Four Seas: multilateral approaches in the Baltic, Black, Caspian & Mediterranean Seas. The EU and Europes Sea Basins, Tallinn (2011)
2. Ince, A., Topuz, E., Panayirci, E.: Principles of Integrated Maritime Surveillance System. Springler, Netherlands (1999)
3. Rasheed, Z., Shafique, K., Yu, L., Lee, M., Ramnath, K., Choe, T.E., Javed, O., Haering, N.: Distributed Sensor Networks for Visual Surveillance. In: Distributed Video Sensor Networks, pp. 439–449 (2011)
4. Battistello, G., Ulmke, M., Koch, W.: Knowledge-aided multisensor data fusion for maritime surveillance. In: Proceedings of SPIE, vol. 8047 (2011)
5. Wang, B., Ye, M., Li, X., Zhao, F., Ding, J.: Abnormal crowd behavior detection using high-frequency and spatio-temporal features. In: Machine Vision and Applications, pp. 1–11 (2011)
6. Aliakbarpour, H., Khoshhal, K., Quintas, J., Mekhnacha, K., Ros, J., Andersson, M., Dias, J.: HMM-Based Abnormal Behaviour Detection Using Heterogeneous Sensor Network. In: Camarinha-Matos, L.M. (ed.) Technological Innovation for Sustainability. IFIP AICT, vol. 349, pp. 277–285. Springer, Heidelberg (2011)
7. Morel, M., Claisse, S.: Integrated System for Interoperable sensors & Information sources for Common abnormal vessel behaviour detection & Collaborative identification of threat (I2C). In: Proc. of the Ocean and Coastal Observation: Sensors and Observing Systems, Numerical Models and Information Systems, Brest, France (2010)

8. Li, X., Xue, Y., Chen, Y., Mali, B.: Context-aware anomaly detection for electronic medical record systems. In: Proceedings of the 2nd USENIX Conference on Health Security and Privacy, p. 8 (2011)
9. Gupta, S., Hossain, L.: Towards near-real-time detection of insider trading behaviour through social networks. Computer Fraud & Security 2011(1), 7–16 (2011)
10. Shi, Y., Tian, Y., Kou, G., Peng, Y., Li, J.: Network Intrustion Detection. In: Optimization Based Data Mining: Theory and Applications, pp. 237–241 (2001)
11. Fok, V., Lingard, D.M.: Using a Genetic Algorithm to Optimise Maritime Surveillance Performed by Space-based Sensors. National Committee for Space Science & National Space Society of Australia, Barry Drive, Australian National University, Canberra, ACT, Australia (2011)
12. Celik, M., Dadaser-Celik, F., Dokuz, A.S.: Anomaly detection in temperature data using DBSCAN algorithm. In: IEEE Innovations in Intelligent Systems and Applications (INISTA), pp. 91–95 (2011)
13. Tan, K.: A multi-agent system for tracking the intent of surface contacts in ports and waterways. Naval Postgraduate School Monterey CA (2005)
14. Jakob, M., Vaněk, O., Urban, Š., Benda, P., Pěchouček, M.: AgentC: agent-based testbed for adversarial modeling and reasoning in the maritime domain. In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems, vol. 1, pp. 1641–1642 (2010)
15. Gupta, K.M., Aha, D.W., Moore, P.: Case-Based Collective Inference for Maritime Object Classification. In: McGinty, L., Wilson, D.C. (eds.) ICCBR 2009. LNCS, vol. 5650, pp. 434–449. Springer, Heidelberg (2009)
16. Nilsson, M., van Laere, J., Ziemke, T., Edlund, J.: Extracting rules from expert operators to support situation awareness in maritime surveillance. In: Proceedings of the Eleventh International Conference on Information Fusion. IEEE (2008)
17. Auslander, B., Gupta, K.M., Aha, D.W.: A comparative evaluation of anomaly detection algorithms for maritime video surveillance. In: Society of Photo-Optical Instrumentation Engineers (SPIE), vol. 8019, p. 3 (2011)
18. Bernon, C., Gleizes, M.-P., Peyruqueou, S., Picard, G.: ADELFE: A Methodology for Adaptive Multi-agent Systems Engineering. In: Petta, P., Tolksdorf, R., Zambonelli, F. (eds.) ESAW 2002. LNCS (LNAI), vol. 2577, pp. 156–169. Springer, Heidelberg (2003)
19. Gleizes, M.-P., Camps, V., Georgé, J.-P., Capera, D.: Engineering Systems Which Generate Emergent Functionalities. In: Weyns, D., Brueckner, S.A., Demazeau, Y. (eds.) EEMMAS 2007. LNCS (LNAI), vol. 5049, pp. 58–75. Springer, Heidelberg (2008)
20. Di Marzo Serugendo, G., Gleizes, M.-P., Karageorgos, A.: Self-organising Software - From Natural to Artificial Adaptation. Springer Natural Computing Series (2011)
21. Ristic, B., La Scala, B., Morelande, M., Gordon, N.: Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction. In: 11th International Conference on Information Fusion, pp. 1–7 (2008)
22. Mano, J.-P., Georgé, J.-P., Gleizes, M.-P.: Adaptive Multi-agent System for Multisensor Maritime Surveillance. In: Demazeau, Y., Dignum, F., Corchado, J.M., Pérez, J.B. (eds.) Advances in PAAMS. AISC, vol. 70, pp. 285–290. Springer, Heidelberg (2010)
23. DCNS: Description of Work. Project I2C (2010)
24. Tsankova, D.D., Georgieva, V.S.: From local actions to global tasks: simulation of stigmergy based foraging behavior. Intelligent Systems, 353–358 (2004)

Control of AGVs in Decentralized Autonomous FMS Based on a Mind Model

Hidehiko Yamamoto and Takayoshi Yamada

Department of Human and Information System, Gifu University, Japan
{yam-h,yamat}@gifu-u.ac.jp

Abstract. This paper describes a system that can efficiently control the motion of automated guided vehicles (AGVs) in decentralized autonomous flexible manufacturing systems (DA-FMS). In particular, a mind model called the minimum unit of the mind (MUM) is proposed. Because of the change of mind characteristic of MUM, the AGVs can efficiently avoid route interferences even when many AGVs are operating in DA-FMS. The components of MUM are two stimulation cells, a box, and an arrow. Using application simulations, it is ascertained that MUM is effective for avoiding AGV route interferences.

Keywords: Production System, FMS, Decentralized Autonomous Systems, AGV, Mind Model.

1 Introduction

Decentralized autonomous systems [1–3] are considered one of the 21st century intelligent systems, and research has been conducted on its applicability in production factories. The authors have found that decentralized autonomous systems can be employed in flexible manufacturing systems (FMS) and have developed decentralized autonomous FMSs (DA-FMSs) [4–6]. The research involved controlling the movement of automated guided vehicles (AGVs) with if–then rules. However, it was difficult to express all possible scenarios occurring in a factory as if–then rules in advance. For example, if the number of AGVs increases and many AGVs are simultaneously gathered at the same place, the AGVs' movements may become restricted, resulting in production stoppage. This indicates that it is quite difficult to achieve motion control with if–then rules for an ideal decentralized autonomous system.

On the other hand, the behavior of human beings does not always follow a set of rules. In addition, their actions are not always based on a set of rules. We occasionally make decisions that are either impulsive or have some unexplained reason. This is a human being! It can be considered that one of the human beings decision makings described above, the decision making not always done by rules, corresponds to the works of mind. The aim of our study is to apply the working of the mind to the AGVs and develop a motion control similar to a human mind for the AGVs in DA-FMS.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 186–198, 2012.

This paper proposes a mind model using which two kinds of minds can be realized: an arrogant mind and a modest mind. In addition, preventing the interference of the AGVs in DA-FMS by efficiently controlling their motion by incorporating the mind model that has the characteristic of change of mind in the AGVs without using if–then rules is described.

Several studies have attempted to build a mind model. One study explained the mind from a psychological viewpoint [7]. Rumelhart described the mind as a connectionism that is related to the recognition function of the brain [8]. This study modularizes the mind using the novel concept of change of mind.

In addition, there are a number of studies on AGV control [9]. For example, Takarazaki et al. proposed avoidance methods for heuristic interference [10],[11]. In these methods, paths that have not experienced interference are determined in advance. The real-world AGVs experience unexpected events such as changes in the motion speed. Conventional methods do not consider the effects of unexpected accidents and production schedules are disrupted when such accidents occur. Watanabe et al. researched methods for avoiding AGV collisions using Q learning or communication protocols [12]. These methods involve the placement of sensors on AGVs and avoid collisions on the basis of sensor detection information. These avoidance pass is chosen one pass and, in the real DA-FMS, just one pass leads to the possibility of an impossible avoidance pass. In addition, this method cannot handle the scenario involving the gathering of AGVs at the same location. Our approach is different from conventional approaches because we focus on the characteristic of change of mind that can be used to prevent interference of AGVs via the online control of the motion of many AGVs.

2 Mind Model

2.1 Basic Model of the Mind

Many AGVs move during the operation of DA-FMS. As shown in Fig. 1, when the AGVs move on the dotted line, there is a possibility of interference because one AGV faces other oncoming AGVs on the same path (route interference). The authors solved the interference problem by the knowledge exchange method [5]. However, to prevent route interference, it was difficult to prepare for all possible scenarios in advance. This required prior knowledge of all situations that can occur in DA-FMS. In this study, efficient ways were evaluated to prevent this route interference by using the mind model that can be used to switch between kinds of minds: (1) a modest mind that gives way and (2) an arrogant mind that moves by force.

A definition of the mind is “a mental operation that integrates the following five aspects of humans—knowledge, emotions, will, memory, and consciousness” In other words, the mind is not easily expressed using equations and its output can be represented using a kind of black boxes.

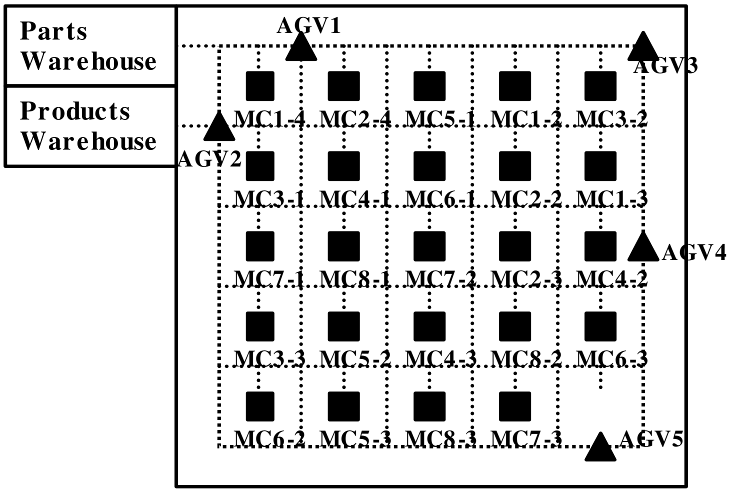


Fig. 1. Decentralized autonomous FMS layout

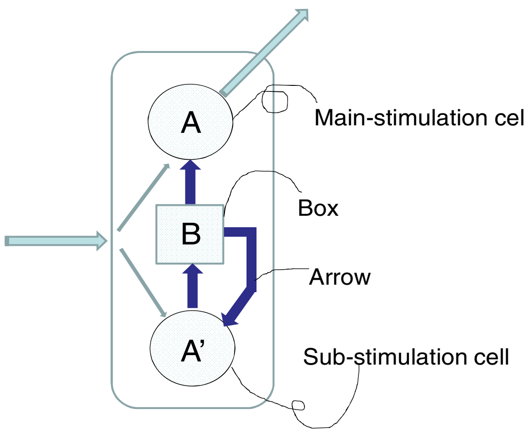


Fig. 2. MUM

In this paper, we propose a mind model that expresses the mind using a kind of black boxes. The model is shown in Fig. 2, and by incorporating the mind model in the AGVs, the problem of preventing interference can be solved.

The three components of the mind model are stimulation cells, a box, and an arrow. As shown in Fig. 2, the mind model is assembled using two stimulation cells and one box. We call this basic mind model the minimum unit of the mind (MUM). MUM represents the smallest unit that exists in the mind. The functions of the three elements of MUM are described below:

[Stimulation cell]. The stimulation cell responds to stimulation inputs called input stimulations that are external to MUM. Depending on the size of the input stimulation, MUM releases stimulation that originates from inside MUM to the outside. We refer to this as the output stimulation. Each MUM has two cells, namely the main stimulation and sub-stimulation cells.

[Box]. The box holds the value called the sending value, and when the box is given the input from the stimulation cell, it outputs the sending value. The value is a negative number.

[Arrow]. The arrow indicates the sending direction of the stimulation or the sending value that is output from the stimulation cell and box.

2.2 Working of MUM

Fig. 2 shows the diagram of MUM; A indicates the main stimulation cell and A' indicates the sub-stimulation cell. After MUM receives one external input stimulation, it determines whether the input stimulation should be transmitted to the main stimulation cell or sub-stimulation cell on the basis of the kind of input stimulation. Finally, the output stimulation from MUM is transferred to the outside. In the case where the selected input stimulations are I(x) and I(y), the output operations are performed by the following procedures.

Step 1: When the input stimulation is received, carry out the following rules.

if the input stimulation is I(x), ***then*** stimulate the main stimulation cell A and go to *Step* 2.
if the input stimulation is I(y), ***then*** stimulate the sub-stimulation cell A' and go to *Step* 3.

Step 2: When the main stimulation cell is in the excited state, MUM outputs the output stimulation. If MUM is in the normal state, go to *Step* 1 after receiving the next input stimulation.
Step 3: When the sub-stimulation cell is in the excited state, MUM transmits the stimulation to the box and goes to *Step* 4. If MUM is in the normal state, go to *Step*1 after receiving the next input stimulation.
Step 4: The box transmits the sending value to the two stimulation cells.
Step 5: Subtract the sending value from the summed value of the two stimulation cells and go to *Step* 1 after receiving the next input stimulation.

Each stimulation cell does not immediately transmit its stimulation after it is received, but there is a time lag before transmission. To generate the time lag, each stimulation has two kinds of values. One is the summed value of the input stimulations input to the stimulation cell and the other is its threshold value. The summed value of the input stimulation is the sum of the values of all received input stimulations. When the summed value of the input stimulation exceeds the threshold value, the output stimulation is transmitted. For example, as shown in Fig. 3, a numerator of 2 corresponds to the summed value of the input stimulation and a denominator of 5 corresponds to the threshold value. If the summed value of the input stimulation is increased by 4, it exceeds the threshold value of 5 and the stimulation cell transmits the output stimulation.

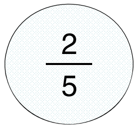


Fig. 3. Stimulation cell of MUM

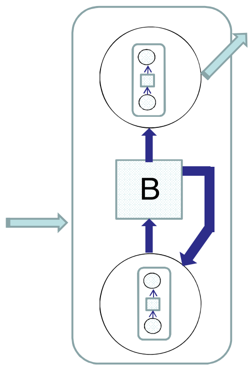


Fig. 4. Nest mind model

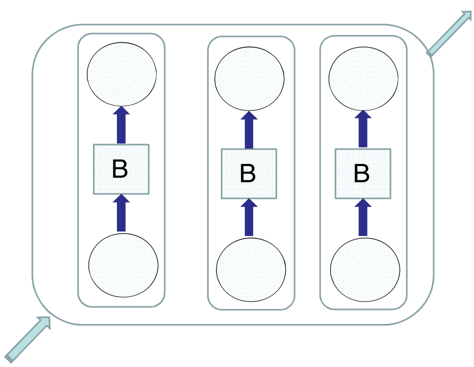


Fig. 5. Parallel mind model

In this way, the stimulation cell maintains two states; that is, the summed value is either greater than or below the threshold value. In particular, MUM whose main stimulation cell's input summed value is equal to or below the threshold value is called the normal state and MUM whose main stimulation cell's input summed value is greater than the threshold value is called the excited state. *Step* 5 in the above procedure can change the state of MUM from the excited state to the normal state. The procedures used two kinds of input stimulations, I(x) and I(y). I(x) is the input stimulation when the main stimulation cell maintains the normal state and I(y) is the input stimulation when the main stimulation cell maintains the excited state. In this way, by expressing the two MUM states (the normal and excited states), our mind model can also switch between the two states. This change is referred to as the change of mind.

Here, the threshold value is not constant but variable, and each mind has a different threshold value, which can cause the change of mind to vary. Therefore, the AGVs with different minds can exist. Using the two stimulation cells in MUM, the time taken to send the output from MUM and the time taken to input from the box can be set differently. This can provide a variety of minds.

MUM is the minimum unit of the mind model. Although one unit can be expressed by one kind of mind, combining different MUMs allows many kinds of mind models to be expressed. That is, by incorporating MUM into the stimulation cell of another MUM, we can express a mind model with many MUMs that fit one another. We call this model the nested mind model, which is shown in Fig. 4. Another model called the parallel mind model in which MUMs are arranged in a row is shown in Fig. 5. In this way, although the inputs are the same, the way in which MUMs are incorporated can determine the expression of the various outputs and mind expressions.

2.3 Incorporating MUM in AGVs

Using MUM, we attempt enabling the efficient control of the AGV motion in DA-FMS by incorporating various minds in the AGVs. That is, using the following method, MUM is incorporated in the AGVs to prevent route interference.

As mentioned above, MUM maintains two states: the normal and excited states. We call the mind in the excited state an arrogant mind and that in the normal state a modest mind. When an arrogant mind is incorporated in the AGV, it is forceful in its motion. On the other hand, when a modest mind is incorporated in the AGV, it gives way to other AGVs.

Therefore, when external stimulations are input to MUM, there can be two kinds of change of mind: from a modest mind to an arrogant mind and from an arrogant mind to a modest mind. Because of the change of mind, the AGV can either give way or forcefully move, hence preventing route interference.

2.4 Receiving and Incorporating MUM Stimulations

To realize an AGV that switches between an arrogant mind and a modest mind using MUM, both input and output stimulations are needed. The input is considered the facing time, which is the time taken for the AGVs to confront each other after starting. For example, if the facing time is set as 3 s and AGVs confront each other after 6 s, the input stimulation is twice the input. The output stimulation is given a value of 1.

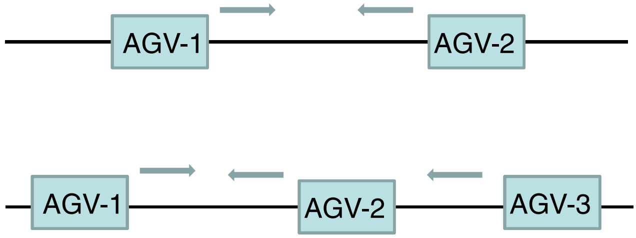


Fig. 6. Example 1 of AGV confrontations

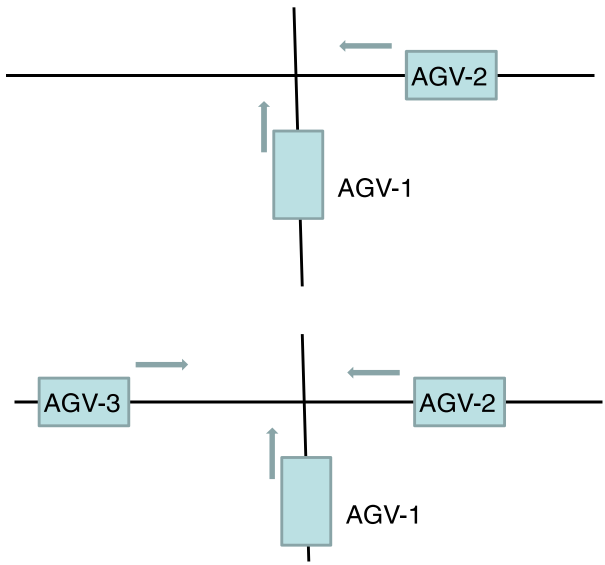


Fig. 7. Example 2 of AGV confrontations

Facing is explained as follows. When there is interference on the AGVs' routes, they might confront each other head-on or sideways. This is called facing. The upper image in Fig. 6 shows the case where two AGVs confront each other head-on. The lower image is the case where two AGVs confront each other and another AGV behind one of the two AGVs blocks the path needed for it to reverse. Fig. 7 shows another case where AGVs are about to collide at an intersection. Generally, the scenarios are not limited to those involving two AGVs and more than two AGVs can be involved in the standoff. Increasing the number of AGVs moving in DA-FMS will increase the occurrence of route interference among the AGVs, and it will become more difficult to determine which AGV should first give way or advance. Using MUM enables controlling the motion in such cases by using the ability of the minds of the AGVs to be changed from modest to arrogant.

3 Simulation Applications

3.1 Avoidance of Route Interference

We simulated DA-FMS on a computer, in which the AGVs incorporated with MUM move. By operating the DA-FMS, we attempted to ascertain the utility of MUM. As shown in Fig. 2, we adopted the simplest MUM as a mind and incorporated it in each AGV. DA-FMS consists of 24 machining centers (MC), 1–6 AGVs, a parts warehouse, and a product warehouse, as shown in Fig. 1. The AGVs moved on a dotted line at a constant speed.

Each AGV has self-knowledge comprising four elements [5]. Self-knowledge is knowledge possessed by an AGV that allows it to move autonomously. The elements include the AGV's name, the coordinates of the intersection passed through last, the coordinates of the intersection that the AGV will move through next, and the coordinates of the intersection two points ahead of the AGV.

It is difficult for the AGVs to move independently if they do not have information of other AGVs. However, it is not feasible for each AGV to hold all information about all other AGVs. To solve this problem, the AGVs are given basic knowledge concerning the current and subsequent paths of other AGVs. This knowledge is called neighbor knowledge and includes the elements of self-knowledge pertaining to the other AGVs. Based on neighbor knowledge obtained from the other AGVs, each AGV recognizes the current location and the next intersection through which the AGV from which the knowledge was obtained will pass. Whenever an AGV passes through an intersection, it sends its self-knowledge to all other AGVs via radio transmission. When an AGV receives the self-knowledge of other AGVs, it stores it as neighbor knowledge.

For example, an AGV detects the possibility of confrontation on the basis of the third and fourth elements of its self-knowledge and neighbor knowledge and merging the two sets of data. Fig. 8 shows an example of such a confrontation involving AGV-1 and AGV-2, where AGV-1 is a modest AGV and AGV-2 is an arrogant AGV. Fig. 8 indicates that the two AGVs have confronted each other between intersections I(1) and I(2) and have stopped in the middle of their paths.

We consider this situation to represent a basic confrontational scenario and the AGVs would take the following interference avoidance precautions. There are five paths that can be taken by the AGVs, namely R(1)–R(5). However, because AGV-3 and AGV-4 are behind AGV-1 in the scenario shown in Fig. 8, the actual paths that can be taken by AGV-1 to prevent a confrontation with AGV-2 are R(2), R(4), and R(5). The three paths behind the modest AGV and the two branching paths ahead of the modest AGV correspond to candidate paths for avoiding confrontations, and we refer to them as avoidance candidate paths (ACPs). All ACPs are not always chosen as actual avoidance paths because other AGVs may be located in one of the ACPs. The actual paths on which an AGV can safely move must be chosen from among ACPs. The chosen path is referred to as the practical avoidance path (PAP).

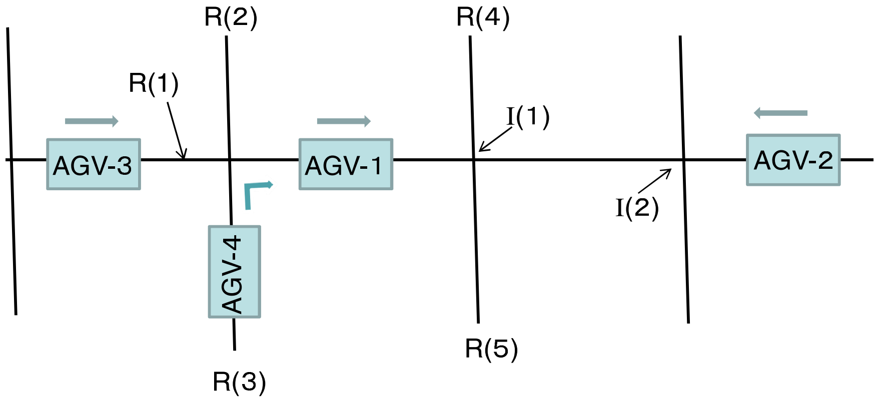


Fig. 8. Example showing possible detours

For a modest AGV, the strategy for avoiding interference involves (1) comparing self-knowledge with neighbor knowledge, (2) searching for PAP from among ACPs, and (3) selecting one path from among PAPs. The selected path corresponds to the path in which the modest AGV actually moves. The strategy is carried out by the following procedure.

1) The modest AGV searches the other AGVs' current locations by checking the second and third elements of its neighbor knowledge to determine whether there are any other AGVs in ACP. If there are none, all ACP paths become PAPs. However, if there are other AGVs in ACP, this path is deleted from ACP and ACP is updated.
2) One path is randomly selected from among ACPs and the modest AGV moves to it.
3) The arrogant AGV moves to the scheduled path.

In this way, after an AGV passes through an intersection, it always determines the possibility of subsequent route interference using its self-knowledge and the neighbor knowledge of the other AGV from which it received information. If there is a confrontation between two AGVs, they experience a change of mind. As a result, the AGV whose mind has become modest moves to the evacuation path using the above strategy.

3.2 Simulation Results

We considered several AGVs for the prototype DA-FMS, and they were all given different minds, as described below. For the initial values of MUM, the summed values of the main stimulation cells and the sub-stimulation cells were set as 0. The threshold values of the main stimulation and sub-stimulation cells were randomly chosen to be within the range of 10–15 and 30–35, respectively. For the sending value of the box, a tentative value was chosen between 1 and the threshold value of the main stimulation cell, and the negative of the tentative value was set as the sending value. The timing of the received input stimulation was the time taken for the AGV to experience a confrontation. Every time an AGV experiences confrontation, an input stimulation of 2 s was set to be received.

In real production conditions, DA-FMS manufactures nine kinds of parts (P_1–P_9). These nine parts are always placed in the parts warehouse and the parts no longer needed for operations are transferred to the product warehouse by an AGV. The nine parts have a preset production ratio of P_1:P_2:P_3:P_4:P_5:P_6:P_7:P_8:P_9 = 5:6:3:3:2:1:4:5:2. Each part is needed for different processes and has different process sequences. For example, as shown in Table 1, P1 needs the machining center sequence $MC_1 \rightarrow MC_2 \rightarrow \ldots \rightarrow MC_8$. The process time for each machining center was also different and for P_1, the process times were 120 s, 150 s,…, 100 s, as shown in Table 1. Similarly the process sequences and process times for parts P4–P9 were unique and were different from those of P_1 and P_2; their values are omitted from Table 1.

Table 1. Machining conditions

Parts	P_1		P_2		P_3	
Production ratio	5		6		3	
Process time (second)	MC_1	120	MC_1	120	MC_2	180
	MC_2	150	MC_3	60	MC_4	180
	MC_3	100	MC_5	180	MC_6	120
	MC_4	60	MC_7	120	MC_8	90
	MC_5	180	MC_2	180	MC_1	180
	MC_6	120	MC_4	150		
	MC_7	180				
	MC_8	100				

When DA-FMS was operated for the first time, the AGVs were randomly placed in it. The machining centers do not have any parts and the initial operation of DA-FMS was such that each AGV started to move to the parts warehouse to retrieve parts.

On a computer, we constructed six types of DA-FMSs and performed production simulations; each DA-FMS had 24 machining centers and 1–6 AGVs. After all six DA-FMS had operated for 8 h during the day, the outputs and the number of confrontations were determined. All the six types of DA-FMS experienced 10 short randomly determined delay intervals, which represented stoppage time for the machines and operation delays that may occur. The production simulations were performed ten times. Table 2 shows a simulation results. In the case where there are two AGVs, ten times the average of the outputs was 85.8 parts and the average number of confrontations was 76.5. The table also shows that the number of confrontations for one AGV was 0 and that for six AGVs was 1176.5. This indicates that the greater the number of AGVs, the greater the number of confrontations, and is an obvious result. During the 8-h operation of DA-FMS, none of the AGVs stopped because of confrontations, and all AGVs could avoid confrontations. In contrast, using the conventional AGV control method employing if–then rules, there were many unexpected AGV stoppages. This result indicates that the control using the rules has limitations. The values in the parenthesis of Table 2 indicate the outputs acquired by the conventional method and demonstrate this limitation. In contrast, the AGVs controlled by MUM moved smoothly and there were no machine stoppages. This proves the utility of MUM.

Table 2. Simulation Results 1 using MUM

Number of AGVs	1	2	3	4	5	6
Average outputs	43 (44)	85.8 (83.88)	122.25 (120.05)	153.05 (83.5)	182.9 (0)	209.2 (0)
Number of confrontations	0	76.5	226.85	496.3	775.95	1176.75

Table 3. Simulation Results 2 using random-1

Number of AGVs	1	2	3	4	5	6
Average outputs	43	85.85	121.8	60	0	0
Number of confrontations	0	71.55	425.9	15695.9	27827.7	28427.8

Table 4. Simulation Results 3 using random-5

Number of AGVs	1	2	3	4	5	6
Average outputs	43	84.45	120.24	125.85	0	0
Number of confrontations	0	162.15	1048.86	5390.05	27960.5	28470.1

Table 5. Simulation Results 4 using random-10

Number of AGVs	1	2	3	4	5	6
Average outputs	43	85.6	119.65	140.8	50.36	0
Number of confrontations	0	260.5	929.7	3150	20504.92	28428.8

The proposed method where the AGV either proceeds forcefully or retreats is considered to involve a random selection between the two actions (move ahead by force or retreat). We refer to this as a random change of mind. Under the same production conditions, we conducted simulations using random changes of mind. The frequencies of random changes of mind were every 1 s (random-1), every 5 s (random-5), and every 10 s (random-10). Tables 3, 4, and 5 show the results for random-1, random-5, and random-10, respectively. In Table 3, ten times the average output with two, three, four, five, and six AGVs were 85.5, 121.8, 60, 0, and 0, respectively. This indicates that the outputs decreased with increasing number of AGVs. On analyzing this decrease, it was found that there were frequent changes of mind. None of the AGVs avoided route interference, and there were cases where the AGVs were stuck and could not proceed any further. As a result, the transport of parts stopped, and the parts were not transported to machining centers for 8 h. In the cases involving five and six AGVs, a gridlock occurred relatively quickly and the transportation of parts stopped for 8 h, preventing any products from being produced, and hence, the output of 0. Similarly, in the cases of random-5 and random-10, a larger number of AGVs results in an output that is lower or 0. The above results are illustrated in Fig. 9 and show that allowing easier random changes of mind leads to a gridlock as the number of AGVs are increased. With MUM, the AGVs could maintain the production schedule without forming gridlocks, and DA-FMS operations were performed on schedule. This proved the utility of MUM.

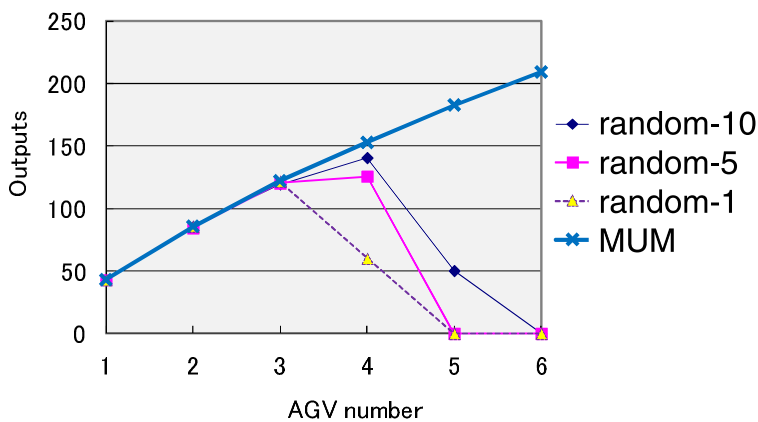


Fig. 9. Outputs

4 Conclusions

This paper describes the control of AGV motion, which is necessary for the efficient operations of DA-FMS. In this control method, a mind is incorporated in the AGV with the aim of preventing a gridlock of the AGVs. A mind model referred to as MUM is used. It comprises three components: the stimulation cell, an arrow and a box. Using MUM, the actions of the AGV mimic two kinds of minds, an arrogant mind and a modest mind, and it is possible to switch between the two minds.

We proved the utility of MUM by simulating the operation of DA-FMS in which there were many AGVs incorporated with MUMs. In our simulation using 1–6 AGVs, there were no AGVs gridlocks during DA-FMS operations. For comparison, production simulations with AGVs whose minds randomly changed were also conducted. The random change of mind resulted in gridlocks, and parts were not transported during the DA-FMS operation, leading to production stoppages. The above results prove the utility of MUM.

Different combinations of the proposed MUM can be used to realize various types of minds. In future, we aim to extend the types of mind that can be realized.

References

1. Coulouris, G., Dollimore, J., Kindberg, T.: Distributed Systems Concepts and Design. Addison-Wesley, Wokingham (2001)
2. Mori, K., Shiibashi, A.: Trend of Autonomous Decentralized System Technologies and Their Application in IC Card Ticket System. IEICE Transactions on Communications E92.B(2), 445–460 (2009)
3. Yamamoto, H., Marui, E.: Decentralized Autonomous FMS Control by Hypothetical Reasoning Including Discrete Simulator. In: Monostori, L., Váncza, J., Ali, M. (eds.) IEA/AIE 2001. LNCS (LNAI), vol. 2070, pp. 571–581. Springer, Heidelberg (2001)
4. Yamamoto, H., Rizauddin, B.R.: Real-time Decision Making of Agents to Realize Decentralized Autonomous FMS by Anticipation. International Journal of Computer Science and Network Security 6(12), 7–17 (2006)
5. Yamamoto, H., Qudeiri, J.A., Jamali, M.A.J.: Real-Time Control of Decentralized Autonomous Flexible Manufacturing Systems by Using Memory and Oblivion. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) KES 2006, Part I. LNCS (LNAI), vol. 4251, pp. 252–259. Springer, Heidelberg (2006) ISBN 978-3-540-46535-5
6. Yamamoto, H., Yamada, T.: Cooperation of AGVs' Head-on Collision Avoidance by Knowledge Exchange in Autonomous Decentralized FMS. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010, Part I. LNCS, vol. 6276, pp. 158–167. Springer, Heidelberg (2010)
7. Nakajima, H., et al.: Social Intelligence in a Human-Machine Collaboration System. Transactions of the Japanese Society for Artificial Intelligence 19(3), 184–196 (2004)
8. Rumelhart, D.E.: The Architecture of Mind: A Connectionist Approach. In: Mind Design II, pp. 204–232. The MIT Press (1997)
9. Krishnamurthy, N.N., et al.: Developing Conflict-Free Routes for Automated Guided Vehicles. Operations Research 41(6), 1077–1090 (1993)
10. Hohzaki, R., et al.: Heuristic Methods for Determining Interference-Free Routes of Automated Guided Vehicles. The Journal of the Japan Society of Mechanical Engineers 57(537), 1763–1769 (1991)
11. Watanabe, M., et al.: Acquisition of a Communication Protocol for Autonomous Multi-AGV Driving. In: The Proceedings of the Second International Conference on Intelligent Processing and Manufacturing of Materials, vol. 2, pp. 1115–1121 (1999)
12. Furukawa, M., et al.: Collision Avoidance with a Moving Object Based on Q-Learning. The Journal of the Japan Society of Mechanical Engineers 69(680), 1051–1057 (2003)

Developing Intelligent Surveillance Systems with an Agent Platform

David Vallejo, Luis M. García-Muñoz, Javier Albusac, Carlos Glez-Morcillo, Luis Jiménez, and Jose J. Castro-Schez

University of Castilla-La Mancha
Paseo de la Universidad 4, Ciudad Real, Spain
David.Vallejo@uclm.es
http://www.esi.uclm.es/www/dvallejo

Abstract. In this work we discuss the advances in the development of an existing agent platform [20] taking in mind the domain of intelligent surveillance. We have paid special attention to key aspects such as scalability, robustness and flexibility. Furthermore, we have focused our effort to i) implement a scalable replication service for the main components of the platform, ii) provide the developers with a flexible agent-based development model and iii) integrate concurrent negotiations. A MAS deployment is also described to monitor an indoor environment.

1 Introduction

Intelligent surveillance can be defined as the application of techniques, algorithms and methods of Artificial Intelligence in order to develop advanced security systems that perform the monitoring tasks traditionally carried out by human operators. Within this context, intelligent surveillance represents a hot research topic, addressing relevant research challenges such as the effective detection, tracking and, from a more general point of view, behavior understanding of moving objects [19].

The ultimate goal of intelligent surveillance consists in automating the monitoring of complex environments in a robust and efficient way. In other words, intelligent surveillance also aims at advancing the state of the art of traditional surveillance systems, where the security personnel are responsible for supervising suspicious activities on monitors. However, the continuous observation of this kind of devices for surveillance purposes has proven to be ineffective after long periods of time [18]. Within this context, intelligent surveillance can reduce this human dependence by correctly assisting the security personnel and optimizing the monitoring process.

Currently, there are artificial systems capable of detecting anomalous behaviors, identifying suspicious or lost objects, analyzing the trajectories of moving objects, or even detecting crowds (see [19] for a thoroughly review).

In the last few years, two of the main factors that have determine the evolution of intelligent surveillance systems [16] are i) the distribution of the sensors

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 199–208, 2012.

used to gather data from the environment (e.g. security cameras) and ii) the fact that our society demands a higher security. On the one hand, the physical distribution of surveillance sensors implies the use of distributed processing and information fusion. On the other hand, a higher security implies the use of Artificial Intelligence techniques and methods to design more sophisticated solutions. These solutions should be are able to generate high-level information when monitoring complex environments.

So, from a general point of view, designing advanced surveillance systems can be understood as designing knowledge-based distributed system. In this way, the information gathered from the environment is used to generate knowledge and such knowledge is used to provide a sophisticated surveillance. Within this context, agent technology represents a more than suitable solution to face this challenge. In fact, multi-agent systems have been widely used to address surveillance-related problems such as image segmentation [3], multi-camera tracking [15], behavior understanding [4], or information fusion [11], to name a few.

Although there exists a number of agent platforms ready-to-use, such as JADE [2], the developers of agent-based surveillance applications usually implement their systems without using these general-purpose agent platforms. In this work, we try to reduce this gap and discuss the software advances we have integrated into our agent platform in order to attract the developers of this kind of applications. In particular, this paper covers three relevant aspects:

1. A **scalable replication service** to increase the system robustness, which is critical in surveillance systems, paying special attention to the basic services of the agent platform.
2. A **flexible agent model** based on the use of behaviors that can be customized by the developer, considering the particular case of parallel behaviors.
3. The integration of **concurrent negotiations** [12] for the agents to gain flexibility (if needed) when monitoring environments.

Although these three aspects can be used to develop any agent-based application that requires them, we claim they are particularly important in the surveillance domain, where scalability, robustness and flexibility are key elements. Our proposal not only aims at guaranteeing these key aspects but also addresses the configuration, development and deployment of agent-based surveillance applications. To do that, the developed agent framework relies on a modern communication middleware [9] that supports a great number of programming languages, operating systems and hardware platforms.

The rest of the chapter is organized as follows. Section 2 positions our research within the context of relevant related works. In Section 3, we present an overview of the agent platform and discuss in detail the three previously mentioned advances to improve the development of agent-based surveillance applications. Section 4 describes a case study where we use the agent platform to monitor a specific environment. Finally, Section 5 concludes and resumes the main ideas of this work and outlines new research lines.

2 Related Work

There are multiple agent development frameworks that can be used to develop agent-based applications. Table 1 shows the most relevant agent platforms and their properties. However, from a general point of view, these platform have not been widely used in the surveillance domain, but there is a number of proposal that follow the guidelines of the FIPA committee to design surveillance agents.

Currently, JADE (*Java Agent DEvelopment Framework*) [2] is probably the most widespread agent framework and has been used to deploy a great number of agent-based applications. JADE is compliant with the set of standards promoted by the FIPA (*Foundation for Intelligent Physical Agents*)[1] committee. From a practical point of view, JADE is organized in two development packages: i) a FIPA-compliant multiagent system and ii) a package for developing Java-based agents. Therefore, the developers that make use of JADE must implement their agents in Java by following the developer guide provided by JADE [2]. In [4], the authors proposed the design of a FIPA-compliant agent to manage and detect network traffic.

In the surveillance domain, this constraint might affect the performance of the developed application and restricts the freedom to use a different programming language such as, for instance, C or C++, which are particularly used in surveillance. In contrast to this approach, our platform relies on the use of ZeroC ICE [9], a modern communication middleware that covers a higher number of programming languages and provide the agent platform with native services that facilitate the development and management of agent-based surveillance systems, such as transparent location service, implicit server activation or load balancing policies.

Jadex is an open-source software framework [14] to design goal-oriented agents that follow the *belief-desire-intention* model. This is a relevant advantage since the BDI model has been widely used for years. The main goal of the project is to develop agent-based systems in an easy and intuitive way by providing a middleware using well-known engineering foundations. Jadex allows developers to build software agents in XML and Java and can be deployed on different middlewares, such as JADE. In [13], the authors used Jadex to implement a number of software agents responsible for analyzing the frames captured by surveillance cameras.

Cougaar [8] is an open-source, Java-based architecture to develop distributed agent-based applications. One of the main objectives of this project is to provide a tool for large-scale and flexible MAS, which is clearly an advantage in surveillance applications where a great number of moving objects must be simultaneously monitored. However, it does not make use of standards so that interoperability with other systems is not achieved.

Another existing agent framework is Agent Factory [17], which can be understood as an extensible framework to develop and deploy agent-oriented applications and aims at giving support to the development of complex distributed

[1] http://www.fipa.org

Table 1. Relevant agent-platforms

Name	Language	Standard	Open Source	Relevant Reference
JADE	Java	FIPA	Yes	[2]
Jadex	Java	FIPA	Yes	[14]
Cougaar	Java	No	Yes	[8]
Agent Factory	Java	FIPA	Yes	[17]
JACK	Java	No	No	[21]

systems in a wide range of application domains. The main characteristics of this framework are the adoption of FIPA standards, the use of a *plug and play* philosophy to deploy and configure multiple agent types, the use of Java, and the inclusion of some agent interpreters.

3 Agent Platform Description

3.1 General Overview

In this work we address how a previously designed agent platform [20] has been extended in order to support some relevant characteristics of agent-based surveillance systems. This agent platform is based on the set of standards defined by the FIPA committee.

This platform is composed of the three basic services: i) the Agent Management System (AMS), which provides the platform with a *white pages* service, ii) the Directory Facilitator (DF), which provides the platform with a *yellow pages* service for the agents to offer their services, and iii) the Message Transport System (MTS), which represents the default communication method between agents.

All these services are persistent and can be easily replicated through multiple physical computers. The platform core is implemented in C++ in order to guarantee efficiency, which is essential in the surveillance domain, and it makes use of the middleware ZeroC ICE for communication purposes. Software agents are implemented in C++ and Python. The source code is available online[2] and it is released under GPL license.

3.2 Service Replication

As previously introduced, the basic services of the agent platform support persistence and replication. On the one hand, persistence relies on Freeze Evictor, a service provided by ICE. On the other hand, replication has been implemented by adopting the mixed model depicted in Figure 1.

[2] `http://www.esi.uclm.es/www/dvallejo/agentplatform/src.tar.gz`

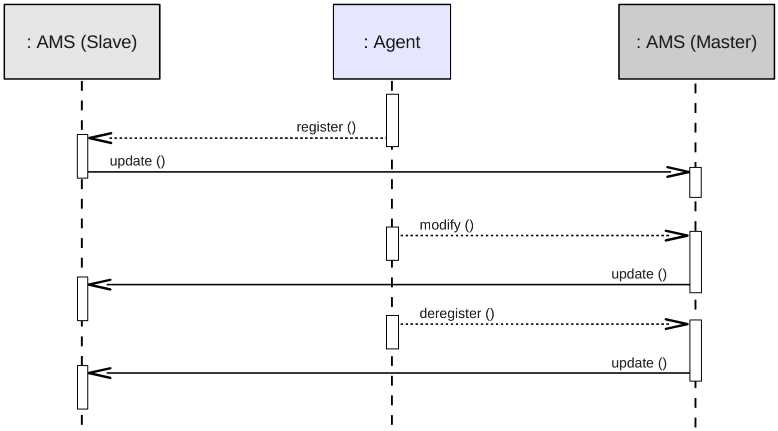


Fig. 1. Simplified graphical view of the replicated AMS service. Interaction between replicas is carried out through specific event channels. Updates are performed by implementing the *Observer* design pattern.

The replication model works as follows:

1. An agent asks a replica of a well-know service, such as the AMS or the DF, for a proxy, which is the representation, on the client side, of the actual service.
2. Depending on the operation type, read-only or update, the replica behaves one way or another.
   (a) For read-only operations, it does not matter whether the queried proxy represents the master or an actual slave (replica) of the service.
   (b) If the service state is to be updated, then two cases are distinguished. If the replica is the master, then the update is carried out and the master distributes the changes to all the slaves. If the replica is a slave, then it notifies the master so that the latter one can initiate the previously mentioned procedure.

When starting the agent platform, the bully algorithm [7] is used in order to determine the master between all the existing replicas. Furthermore, all the core services support a journaling system so that they can be synchronized in every moment. Basically, the journaling system commits all the changes, using the service Freeze Map provided by ICE. In case of error, the journaling system is able to recover the latest actions of a particular service and undo the error cause. Once again, this characteristic is particularly relevant in the surveillance domain.

It is important to remark that persistence and replication are not required by the Message Transport System, for this service is only an intermediary between agents and does not maintain an internal state.

Finally, the location of all the software entities within the platform is transparently provided by a transversal service, named *locator*, of the middleware ZeroC ICE. Fortunately, this service can be also replicated.

In an agent-based surveillance system, the advantages of persistence and replication are obvious, since they allow the agents of the system to keep on communicating even when the nodes fail. However, the developers are responsible for distributing the replicas between the physical nodes, that is, they have to explicitly define how the replicas will be deployed.

3.3 Agent Model

The standard execution flow of the basic agents provided by the agent platform is inspired in the behavior-oriented model of JADE [2], although we have implemented our own model. Basically, the behaviors represent tasks to be performed by the agents (see Figure 2). Two general behaviors can be distinguished:

- Simple, for behaviors that are not further divided into other behaviors. Similarly, simple behaviors can be classified as follows:
  - Cyclic, which are continuously executed until the agent is destroyed.
  - One Shot, which are executed one single time.
- Composite, for behaviors that are composed of other behaviors. Similarly, simple behaviors can be classified as follows:
  - Sequential, so that all their sub-behaviors are sequentially executed.
  - Parallel, so that all their sub-behaviors are concurrently executed.

The execution of behaviors in our agent platform slightly differs from JADE. Our main contribution is that parallel behaviors are fully parallel because they run in

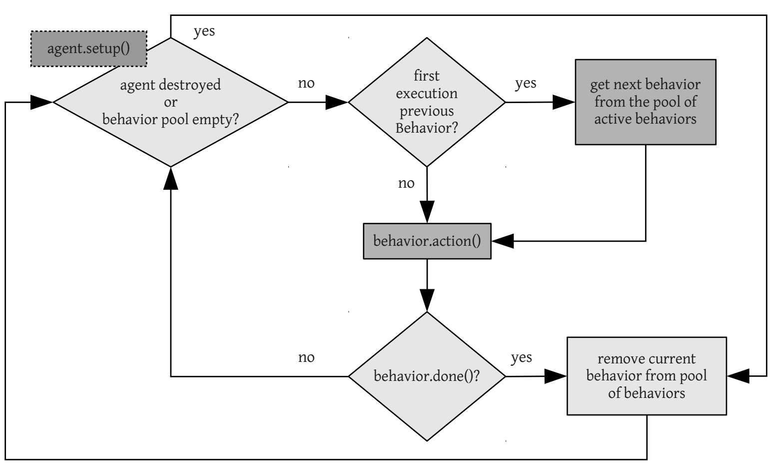


Fig. 2. Control flow of the software agents

```
class BehaviourThread:
 public IceUtil::Thread
{
  private:
    bool _started;
    Behaviour* _behaviourToRun;
    vector<bool>* _donePool;
    int _doneN;

  public:
    BehaviourThread(Behaviour*,
                    vector<bool>*, int);

    virtual ~BehaviourThread();
    virtual void run();
};
```

```
class ParalellBehaviour:
 public CompositeBehaviour
{
  private:
    IceUtil::Mutex _m;
    bool _started;
    vector<Behaviour*> _behaviourPool;
    vector<bool> _donePool;
    bool _allDone;
    int _actualBehaviour;
    vector<IceUtil::ThreadControl>
     _threadPool;

  public:
    ParalellBehaviour();
    virtual ~ParalellBehaviour();
    virtual void action();
    virtual bool done();
    virtual void reset();
    void addSubBehaviour(Behaviour*);
};
```

```
class DataLoggerBehaviour:
 public FIPA::Behaviours::CyclicBehaviour
{
  private:
    IceUtil::ThreadControl _thisThread;
    FIPA::AgentPrx _centralServer;
  public:
    DataLoggerBehaviour(const FIPA::AgentPrx&);
    ~DataLoggerBehaviour();
    std::string generateContent() const;
    void action();
};
```

```
class DataLogger: public FIPA::AgentI
{
  private:
    Ice::CommunicatorPtr _communicator;
    FIPA::AgentPrx _centralServer;

  public:
    DataLogger(Ice::Identity,
               Ice::ObjectAdapterPtr);
    ~DataLogger();

    void setup();
    void takeDown();
};
```

Fig. 3. Some relevant part of the source code of the agent platform (parallel behaviors and definition of a domain-specific agent)

different threads. In fact, our platform makes extensive use of threads. Figure 3 shows some relevant parts of the source code related to parallel behaviors.

Since we provide agent implementations in C++ and Python, behaviors can be implemented in these two programming languages as well. For the developer of agent-based surveillance system, this approach represents an important advantage over other existing agent platforms. Python can be use to develop rapid prototypes while C++ can be employed to develop the final system.

3.4 Concurrent Negotiations

The agent platform provides an experimental support for concurrent negotiations [12], which are implemented by means of parallel behaviors. Negotiations can be very useful in the surveillance domain in order to devise an architecture based on software agents that provide surveillance services and entities that need such services. Currently, the buyer, the seller and the commitment manager are implemented through cyclic behaviors whereas the negotiator and the buyer are instances of the agent provided by the platform.

4 Experimental Results

In order to demonstrate the application of the agent platform in the surveillance domain, a simple MAS has been implemented to monitor an indoor environment

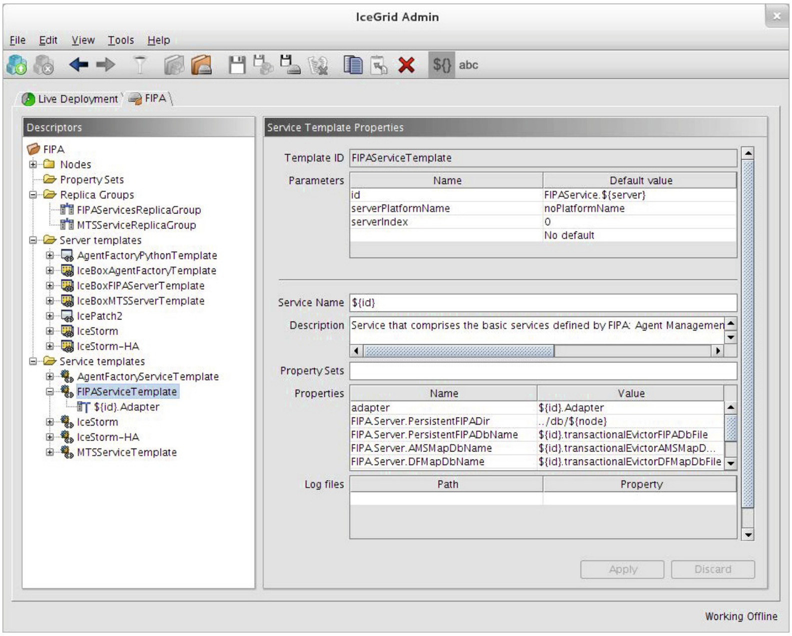


Fig. 4. Screenshot of the graphical user interface employed to configure and deploy MAS by means of the agent platform

where multiple data loggers have been deployed. Each data logger is able to gather information from a light sensor (LDR), a temperature sensor (LM35), a presence sensor (PIR), and a microphone.

A single software agent is responsible for managing one data logger. We have also considered a supervisor, named *central node*, which gets information from the previously mentioned agents and is able to send responses to the agents if they are required (such as resuming the monitoring process, turning off a specific sensor or even specifying threshold for the event generation). Figure 3 shows the definition of the data logger agent, which has been implemented in C++.

It is really easy to model this functionality by using the platform agents and extending cyclic behaviors. The developed agents only have to implement the *action()* method, which determines what to do when executing the cyclic behavior.

The communication between the agents and the supervisor is carried out through event channels, which represent a communication abstraction provided by the agent platform. Basically, the event channel is an implementation of the *publish-subscribe* pattern and it guarantees the independence between information producers and consumers. In other words, it allows data-driven communication between agents.

The administration, configuration and deployment of MAS developed by using the agent platform is straightforward thanks to the graphical tool integrated into the communication middleware (see Figure 4).

5 Conclusions and Future Work

Nowadays, intelligent surveillance is a hot research topic that covers a significant number of areas, ranging from multi-camera tracking to event and behavior understanding. The inherently distributed nature of intelligent surveillance systems makes that agent technology is an ideal candidate to address some of the current challenges. That is our motivation to keep on working on an agent platform that gives support to developers when dealing with domains that demand scalability, robustness, efficiency, and flexibility, such as the surveillance domain.

In this work, we have discussed our recent contributions in order to provide a more scalable and robust agent platform. Within this context, we developed a flexible agent model that allows developers to implement different types of behaviors, paying special attention to parallelism which is extensively supported through the platform. Implementing domain-specific agents is quite straightforward since developers only have to specialize behaviors and agents, defining the code of a few methods that determine how they behave according to the problem to be solved.

Currently, we are working on a multi-agent architecture in order to monitor more complex environments. In particular, the devised architecture will allow the deployment of software agents attached to the moving objects detected in the tracking stage. Each one of then will implement specific behaviors to monitor different events or aspects of interest, such as trajectories, speed, pedestrian crossings, etc. A more complex problem will permit us to improve the and detect new functionalities to be integrated into the agent platform.

Acknowledgements. This work has been funded by the Regional Government of Castilla-La Mancha (Research Project PII1C09-0137-6488) and by the Ministry of Science and Innovation (Research Project TIN2009-14538-C02-02).

References

1. Albusac, J., Vallejo, D., Castro-Schez, J.J., Remagnino, P., Glez-Morcillo, C., Jimenez, L.: A Knowledge-Driven Approach Based On Intelligent Agents to Monitor Complex Environments. IEEE Intelligent Systems 25(3), 24–31 (2010)
2. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing Multi-agent Systems with JADE. Springer (2007)
3. Bovenkamp, E.G.P., Dijkstra, J., Bosch, J.G., Reiber, J.H.C.: Multi-agent segmentation of IVUS images. Pattern Recognition 37(4), 647–663 (2004)
4. Chen, B., Cheng, H.H., Palen, J.: Integrating mobile agent technology with multi-agent systems for distributed traffic detection and management systems. Transportation Research Part C 17(1), 1–10 (2009)

5. Dee, H.M., Velastin, S.A.: How close are we to solving the problem of automated visual surveillance? Machine Vision and Applications 5(19), 329–343 (2008)
6. Detmold, H., Van den Hengel, A., Dick, A., Falkner, K., Munro, D.S., Morrison, R.: Middleware for Distributed Video Surveillance. IEEE Distributed Systems Online 9(2), 1–11 (2008)
7. García-Molina, H.: Elections in a distributed computing system. IEEE Transactions on Computers 100(1), 48–59 (1982)
8. Helsinger, A., Wright, T.: Cougaar: A Robust Configurable Multi Agent Platform. In: IEEE Aerospace Conference, pp. 1–10 (2005)
9. Henning, M.: A new approach to object-oriented middleware. IEEE Internet Computing 8(1), 66–75 (2004)
10. Hu, W., Xie, D., Tan, T.: A hierarchical self-organizing approach for learning the patterns of motion trajectories. IEEE Transactions on Neural Networks 15(1), 135–144 (2004)
11. López-López, L.M., Albusac, A., Castro-Schez, J.J., Jiménez, L.: Semantics-Provided Environment Views for Normality Analysis-Based Intelligent Surveillance. In: International Conference on Agents and Artificial Intelligence, pp. 1–10 (2009)
12. Nguyen, T.D., Jennings, N.R.: Managing commitments in multiple concurrent negotiations. Electronic Commerce Research and Applications 4(4), 362–376 (2006)
13. Patricio, M.A., Carbó, J., Pérez, O., García, J., Molina, J.M.: Multi-Agent Framework in Visual Sensor Networks. EURASIP Journal on Advances in Signal Processing 2007(1), 1–21 (2007)
14. Pokahr, A., Braubach, L., Lamersdorf, W.: Jadex: A BDI Reasoning Engine. Multiagent Systems Artificial Societies and Simulated Organizations, vol. 15, pp. 149–174. Springer (2005)
15. Remagnino, P., Shihab, A.I., Jones, G.A.: Distributed intelligence for multi-camera visual surveillance. Pattern Recognition 37(4), 675–689 (2004)
16. Remagnino, P., Velastin, S.A., Foresti, G.L., Trivedi, M.: Novel concepts and challenges for the next generation of video surveillance systems. Machine Vision and Applications 18(3), 135–137 (2007)
17. Ross, R.J., Collier, R., O'Hare, G.M.P.: AF-APL – Bridging Principles and Practice in Agent Oriented Languages. In: Bordini, R.H., Dastani, M.M., Dix, J., El Fallah Seghrouchni, A. (eds.) PROMAS 2004. LNCS (LNAI), vol. 3346, pp. 66–88. Springer, Heidelberg (2005)
18. Smith, G.: Behind the screens: Examining constructions of deviance and informal practices among cctv control room operators. UK Surveillance and Society 2(2), 376–395 (2004)
19. Valera, M., Velastin, S.A.: Intelligent distributed surveillance systems: a review. Image and Signal Processing 152(2), 192–204 (2005)
20. Vallejo, D., Albusac, J., Mateos, J.A., Glez-Morcillo, C., Jimenez, L.: A modern approach to multiagent development. Journal of Systems and Software 83(3), 467–484 (2010)
21. Winikoff, M.: JACK TM intelligent agents: An industrial strength platform. Multiagent Systems, Artificial Societies, and Simulated Organizations, vol. 15, pp. 175–193. Springer (2006)

Agent-Based Control of Self-sustained Oscillations in Industrial Processes: A Bioreactor Case Study

Piotr Skupin and Mieczyslaw Metzger

Faculty of Automatic Control, Electronics and Computer Science,
Silesian University of Technology,
ul. Akademicka 16, 44-100 Gliwice, Poland
{piotr.skupin,mieczyslaw.metzger}@polsl.pl

Abstract. Application of the Agent and Multiagent Systems (AMAS) in the industrial continuous processes can be a quite interesting and effective solution, especially for monitoring and controlling purposes of bioreactor systems. In the classical approach, a process operator controls the process, but sometimes must take some essential decisions concerning the choice of control strategy. In the case of biological processes, due to their highly nonlinear nature, this can be quite difficult task. For instance, the oscillatory behavior of the bioreactor may lead to higher or lower average biomass concentrations. Hence, there is a need to support the operator by measuring and controlling some additional parameters and this cannot be achieved using only measuring devices and classical control algorithms alone. Based on the agent technology, it has been shown that it is possible to support the operator and to achieve process goals.

Keywords: Agent and Multi Agent System, agent-based process monitoring, bioreactor, self-sustained oscillations.

1 Introduction

In the last decade, there has been significant progress in the theory of agent and multiagent systems (AMAS). According to the first papers in this field [1-3], agent applications were defined as applications operating both independently and collaboratively in a group to exchange information between each other and to attain a common goal [4]. Such agent-based solutions can work quite efficiently in industrial systems, especially, in the field of manufacturing systems. For these systems the application of agent technology can bring significant advantages in comparison to the commonly used information and control systems. For instance, the application of agents based on the concept of holons, works well in the field of Holonic Manufacturing Systems, which provide a new way for more flexible control of manufacturing systems [5-7]. In turn, the continuous processes usually do not involve the agent technology, thus there are very few papers about applications of agents and MASs in this field [8]. Most of the continuous processes are usually based on the classical approach, in which two layers of direct and supervisory control can be

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 209–218, 2012.

distinguished. The layer of direct control is strongly coupled with the supervisory layer and comprises actuators, sensors, controlled plants and controllers. In this layer, the PID and predictive algorithms are the predominant control algorithms [9], which ensure the control of the key process variables. In turn, the supervisory layer is responsible for monitoring and supervision of the whole process. In the supervisory layer, depending on the case under consideration, the process operator can make some decisions concerning changes in the set point values for controllers or even in control strategy. However, in many cases, it requires from the operator sufficient knowledge about the process or a support from a team of qualified experts [10]. Especially, it concerns biological processes [8] where, in order to take appropriate decisions, it is often necessary to know some additional parameters (e.g. average values of microorganism concentration and oxygen mass transfer coefficient [11]) or to detect oscillatory behavior. This information cannot be obtained using only measurement devices alone. Moreover, apart from maintaining key process parameters (e.g. temperature or pH) at desired values, there is often a need to influence the biological process in other way. For example, it might be necessary to change the composition of substrates added to the system. And this also cannot be done using only the classical PID controllers alone. Hence, in order to support the operator when making decisions, it is proposed a system composed of several independent, cooperating with each other and making individual decisions units called agents. The proposed system is tested based on the example of a simple biological process (bioprocess), which takes place in the continuous stirred tank bioreactor (CSTB). Hence, in this paper, most of the attention is given to the description of "agentified" process and these process features, which must be monitored and diagnosed before making the final and fixed (by agents) decision concerning the control strategy. The decision is rather a support (suggestion) for the process operator (because of its consequences for the process). However, the taken decisions can also be realized instantaneously without interference of the process operator.

The paper is organized as follows; the next section presents the description of the bioprocess under consideration with its mathematical model and goals to be attained by the multiagent system (MAS). The detailed descriptions of the proposed agent applications are given in the third section. In turn, in section four, based on the simulation runs the effectiveness of the MAS is presented.

2 Description of the System

The simplest bioprocesses include some initial quantity of microorganisms (biomass concentration) X, which consumes a substrate (of concentration S) fed into the CSTB at the constant flow rate F. In many cases, for such systems, the most common and most essential phenomenon is an occurrence of a stable limit cycle, i.e. self-sustained oscillations (SSO) of biomass concentration resulting from the nonlinear nature of the process [12], [13]. The SSO usually occurs spontaneously for a range of flow rates F or dissolved oxygen levels in the CSTB [14]. It turns out that the operation in the range of SSO may lead to higher (or lower) average values of biomass concentration in comparison to the results obtained in steady states [15]. It should also be emphasized that it is not an easy task to assess the benefits resulting from the oscillatory operating regime, furthermore, there is a need to eliminate oscillations in the final stage of the process (e.g. by using a system of surge tanks). Therefore, the operation of the CSTB in the range of SSO is usually avoided in practice. Hence, depending on the case under consideration, there emerges a need for control of the oscillatory behavior (induction or elimination of the SSO) proceeded by detection of the oscillations. In turn, in the case of enhancing bioreactor productivity, it is necessary to measure the average biomass concentrations on-line, especially in the range of SSO.

In most cases, the choice of an appropriate mode of operation of the CSTB is connected with the choice of appropriate values of the manipulating variables such as: flow rate F, inlet substrate concentration S_{in} or dissolved oxygen concentration by changing the agitation speed of the reactor content. In the presented paper, we apply a method of influencing SSO by feeding the bioreactor with a mixture of appropriately chosen substrates [10].

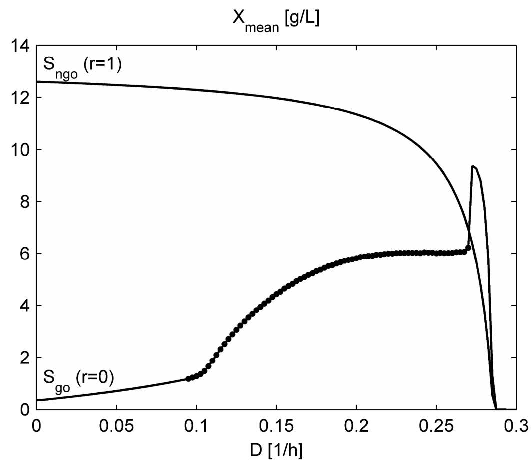


Fig. 1. The average biomass concentration X_{mean} as a function of dilution rate D for two different substrates (S_{go}, S_{ngo})

Hence, we will distinguish a substrate generating SSO (S_{go}), i.e. a substrate for which there exist SSO for some range of flow rates F and a substrate not generating SSO (S_{ngo}), i.e. a substrate for which there are no SSO for any value of flow rate F (Fig.1). By setting appropriate contribution of the individual substrates to the mixture it is possible to determine a desirable operating regime of the CSTB (operation in steady states or in the range of SSO). Hereafter, it is assumed that the CSTB is fed with mixture of two substrates (S_{ngo} and S_{go}) at a constant flow rate F. The contribution of the individual substrates in the mixture is represented by parameter $r \in [0,1]$, i.e. the substrate S_{go} of inlet concentration S_{in1} is fed into the CSTB at a flow rate (1-r)F and the substrate Sngo of inlet concentration S_{in2} is fed at a flow rate rF (Fig.2).

For the system, as shown in Figure 2, a well-known mathematical model with the specific growth rates described by the Monod relations [16] and with the weight coefficients (being proportional to the flow rates (1-r)F and rF) proposed by Metzger in [17] is as follows:

$$\frac{dS_1}{dt} = D\big((1-r)S_{in1} - S_1\big) - \frac{1-r}{\alpha + \beta S_1} \cdot \frac{\mu_{m1} S_1}{S_1 + K_{s1}} X \tag{1}$$

$$\frac{dS_2}{dt} = D\big(rS_{in2} - S_2\big) - \frac{r}{Y_2} \cdot \frac{\mu_{m2} S_2}{S_2 + K_{s2}} X \tag{2}$$

$$\frac{dX}{dt} = -DX + (1-r)\frac{\mu_{m1} S_1}{S_1 + K_{s1}} X + r\frac{\mu_{m2} S_2}{S_2 + K_{s2}} X \tag{3}$$

where: S_1, S_{in1} – are the outlet and inlet concentrations for the substrate S_{go}, S_2, S_{in2} – are the outlet and inlet concentrations for the substrate S_{ngo}, X – biomass concentration, D – dilution rate (D=F/V, F – flow rate, V – volume of the bioreactor medium and V=const), μ_{m1}, μ_{m2} – maximum specific growth rates, K_{s1}, K_{s2} – half saturation constants, $Y_1=\alpha+\beta S_1$ – variable yield coefficient for substrate S_{go} (α, $\beta>0$ – constant parameters), Y_2 – constant yield coefficient for substrate not generating SSO (S_{ngo}).

All the parameter values have been taken from [15]. The proposed model assumes that the most preferable substrate by microorganisms is the substrate, whose contribution to the mixture is the highest.

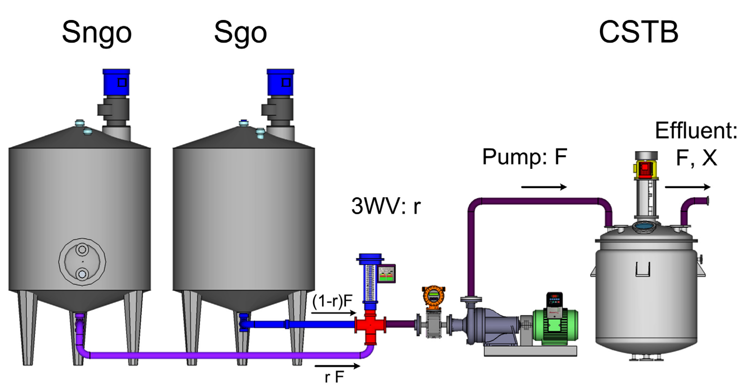


Fig. 2. The scheme of the system with the CSTB fed with the mixture of substrates. The contribution of the individual substrates is set by means of a three port valve (3WV).

Figure 1 presents the average biomass concentrations versus dilution rate D for the two extreme cases: r=0 – the CSTB is fed only by the substrate generating SSO (S_{go}), r=1 – the CSTB is fed only by the substrate not generating SSO (S_{ngo}). The black thick points indicate the range of the SSO. It should be noted that if the dilution rates are too high then the whole culture of microorganisms is washout from the reactor and the biomass concentration tends to zero.

For the bioprocess described by equations (1)-(3) a system of cooperating agents for detection of the SSO, for on-line measurements of average biomass concentration and for choice of the operating regime of the CSTB has been proposed. The detailed description of the MAS has been presented in the next section.

3 The MAS for the CSTB

In the proposed solution two monitoring agents (Agent SSO and Agent Xmean) and one control agent (Agent r) will be distinguished. The monitoring agents are responsible for detection of the SSO and for on-line measurements of the average biomass concentration. The control agent is responsible for elimination of the SSO (determination of value of r). The general scheme of the MAS has been presented in Figure 3.

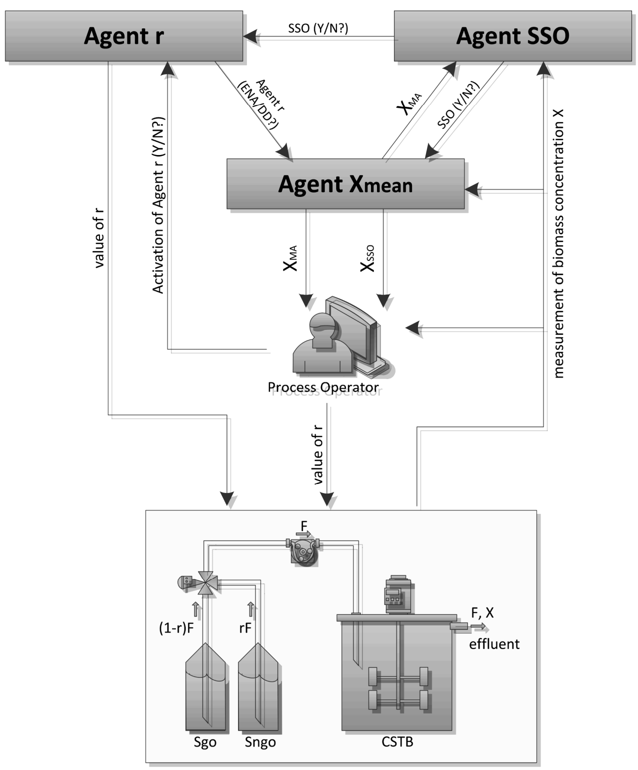


Fig. 3. The general scheme of the MAS comprising three cooperating agents and the process operator

Agent Detecting SSO – Agent SSO

The detection of the SSO is based on the average values of biomass concentration provided by Agent Xmean. A key action of the Agent SSO is the observation of the variable $X^*(t)=X(t)-X_{MA}(t)$, where X(t) – is the output biomass concentration and $X_{MA}(t)$ – is the average biomass concentration calculated by Agent Xmean using moving average (MA). If the SSO are present in the bioprocess, then X*(t) oscillates around zero and the Agent SSO counts the number of zero crossings. The SSO are detected in the system if the number of zero crossings is equal to the value specified by user. The number of zero crossings should be set in accordance with the process under consideration in order to eliminate the detection of the damped oscillations that might appear after step changes in dilution rate D.

Moreover, the maximum and minimum values of X*(t) are calculated on-line in a time window T and, in the case of occurrence of the SSO, $\max(X^*(t))>0$ and $\min(X^*(t))<0$ for $t\in T$. In turn, if the SSO are damped out then the Agent SSO detects the absence of oscillations if the following inequalities are fulfilled: $\max(X^*(t))<\varepsilon$ and $\min(X^*(t))>-\varepsilon$ for $t\in T$, where $\varepsilon>0$ is an appropriately chosen value (in our case $\varepsilon=0.07$). The ε parameter is introduced in order to accelerate the action of Agent SSO, because after disappearance of the SSO the maximum and minimum values of X*(t) do not change their signs for a long time. Hence, the ε parameter introduces some error, but the response of the Agent SSO is much faster. It is worth mentioning that the Agent SSO is also able to detect oscillations resulting from periodic changes in the manipulated variables. Then, the Agent r receives information about the occurrence of the SSO and starts increasing the value of r, which in turn will not eliminate oscillations. In the proposed solution, this case is not taken into account – this would require further modifications to the agent application. It should also be noted that if the dilution rates are quite small (e.g. D=0.01[1/h]), then the response of the bioprocess to changes in D takes more time and detection of the absence of the SSO may occur with considerable delay. However, in practice, dilution rates D are usually large enough.

Agent Attenuating SSO – Agent r

In the proposed solution, the main task of the Agent r is to eliminate the SSO of biomass concentration by increasing the contribution of the substrate Sngo to the mixture. The agent is activated by the process operator, who determines whether the operation of the CSTB in the range of SSO is acceptable or not. If the SSO are acceptable, then the Agent r does not interfere in the process. In the second case, after receiving information about the occurrence of the SSO, Agent r starts increasing the value of r by Δr every Δt interval, where Δt is a multiple of the sampling period of the monitored data. The Agent r increases r until the Agent SSO sends information about the absence of the oscillations. It should be noted that the smaller the value of Δr, the longer the oscillations will persist. However, too large values of Δr can cause unnecessarily high contribution of the substrate Sngo to the mixture after disappearance of the SSO. This may be due to the sluggish response of the process to changes in r and then after the time interval Δt the Agent r may still increase r by Δr. In the event that the process operator significantly reduces dilution rate D, then the value of r may be too large, but it can be manually changed by the operator (Fig.3).

Agent Calculating the Average Biomass Concentration – Agent Xmean

Depending on whether the SSO are detected or not two different scenarios can be distinguished. If the Agent r is enabled (the SSO are not acceptable), then the Agent Xmean only calculates the average biomass concentration using MA ($X_{MA}(t)$) for a sufficiently long time window. If the SSO are present in the system, then the values of $X_{MA}(t)$ oscillate, as well. The oscillations of $X_{MA}(t)$ can be reduced by lengthening the time window for calculating the MA of biomass concentration, but this would cause a significant delay in the calculation of the average biomass concentrations $X_{MA}(t)$ after frequent changes in dilution rate D. Hence, in the second scenario, if the Agent Xmean receives information about the detection of the SSO, then it also calculates the average biomass concentration as the average value for the period ($X_{SSO}(t)$). Similarly, as in the case of detection of the SSO, the Agent Xmean observes the variable $X^*(t)=X(t)-X_{MA}(t)$ and counts the number of zero crossings for every rising edge of $X^*(t)$. This allows for calculating the exact average values of biomass concentration, as an average for the period.

4 Simulation Results

All the obtained results are based on simulation runs of the model described by (1)-(3).

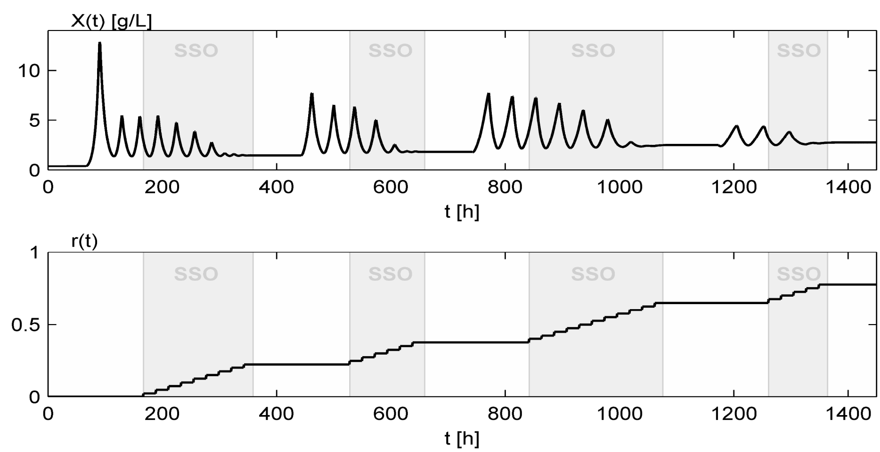


Fig. 4. Time courses of biomass concentration X(t) and parameter r(t). The grey shaded regions present the time intervals of detected oscillations by Agent SSO.

The equations (1)-(3) are integrated using Runge-Kutta (RK4) method with a step h=Dt/NIS, where Dt is an observation step and NIS is a number of integration steps in Dt interval [18], [19]. This means that the process variables are "sampled" every observation step Dt. The process simulator can be treated as a virtual plant for simulation studies, if it is realized in real-time mode in the LabVIEW environment. Hence, the first version of the previously described MAS is also implemented in the LabVIEW environment, in which the agent applications can use the OPC (OLE for

Process Control) communication based on XML (Extensible Markup Language). At the beginning, the first scenario is taken into account, i.e. the operation of the CSTB in the range of the SSO is forbidden and the task of the Agent r is to eliminate oscillations after their detection. The process operator changes dilution rate D, and as a result, for some value of D, the SSO occur. The Agent SSO detects the oscillations, which is presented as a grey shaded region in Figure 4. The information about the detection of the SSO is sent to the Agent r, which starts increasing r by a small step Δr to eliminate oscillations of the biomass concentration. The r parameter is increased until the Agent SSO sends information about the absence of the oscillations. Further changes in dilution rate D may cause the next induction of the SSO, which are detected and eliminated by the Agent r again. Figure 5a presents the regions of the steady states, the SSO and the washout. It is clearly seen that every time after the detection of the SSO, the Agent r increased r parameter to leave the region of oscillations. In turn, the second scenario allows for the operation of the CSTB in the range of the SSO and the Agent r is disabled. In that case, apart from providing information about the average values of biomass concentration $X_{MA}(t)$, the Agent Xmean calculates the average biomass concentration as an average value for the period ($X_{SSO}(t)$) after detection of oscillations by the Agent SSO. Figure 5b presents the average biomass concentrations $X_{MA}(t)$ (black dashed line) and $X_{SSO}(t)$ (black thick line) calculated by the Agent Xmean.

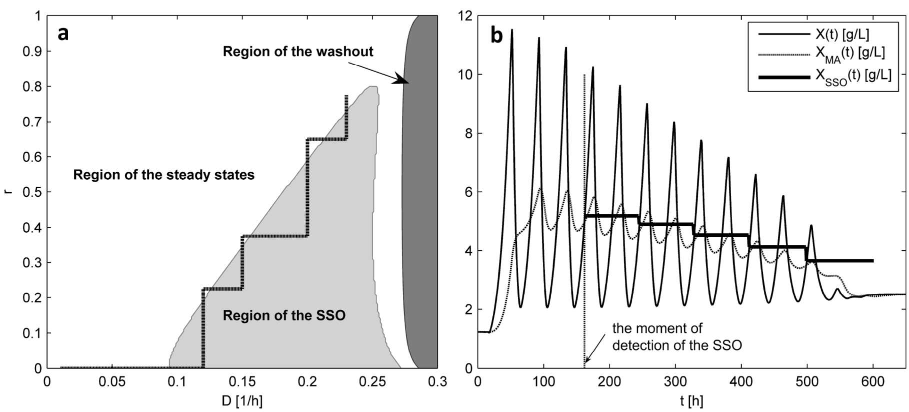


Fig. 5. (a) The regions of the steady states, SSO and washout on the parameter plane (D,r). The black broken line indicates the changes in D made by process operator and the changes in r made by Agent r to eliminate oscillations. (b) Time courses of the biomass concentration X(t) and the average biomass concentrations $X_{MA}(t)$ and $X_{SSO}(t)$.

5 Concluding Remarks

The paper focuses on an accurate presentation of how a few independent, autonomous and cooperating with each other programming entities (i.e. agents) can effectively support a process operator in the bioreactor control system. Many of the papers on agent systems describe different features of MASs, but most of them are out of touch with industrial reality. Only a few of them describe the application of simple (usually single) agents supporting the control systems for continuous processes, and in many cases, the agent applications are written in general-purpose programming languages [8]. Moreover, it is often emphasized that the agent technology is particularly useful, when the application of classical continuous or discrete control algorithms is difficult [8]. Therefore, one of the main objectives of our research was to show that only agent applications can realize non-standard control algorithms for biological processes.

However, when a concrete system is given, the question arises as to whether the proposed system of autonomous and cooperating with each other programming entities is an agent system. The answer to this question should depend on the functionality of the whole system and not on the applied agent platform. In our case, one can refer to many well-known papers in the field of agent systems. For instance, Wooldridge states: “First, any control system can be viewed as an agent” (page 16 in [3]), because it has an autonomy, which is a typical feature of agent applications. The further development of agent technology revealed another desirable feature – the ability of cooperation between several agents (see e.g. [3], [20]). Of course, the proposed MAS has both these features.

These processes, for which experimentation is very expensive and requires a lot of preparation (e.g. biotechnological processes, where the preservation of culture of microorganisms can be a difficult task) the introduction of any new control strategy must be preceded by a thorough analysis. Hence, in our case, a multi-level implementation of the proposed system has been applied. First, we carried out simulation investigations of the agent system and the bioreactor model, which is the cheapest and safest way to study the performance of the whole system. The obtained results showing the oscillatory behavior, but also details of the most important algorithm for detection of the SSO have been described in sections 3 and 4. For simulation purposes, the virtual process was realized in the LabVIEW (a general-purpose programming environment). The first version of the described agent system was also realized in the LabVIEW environment, where the single agents are autonomous programming entities and are capable of using the OPC communication based on XML.

Detailed studies confirmed the possibilities of induction or elimination of the SSO in bioreactor. Because, the obtained results are of great practical importance, they have been presented in our paper.

In the further stages of this research, it is planned to test the MAS for the virtual bioprocess with the use of own (created in Silesian University of Technology [21]) OPC agent applications implemented in JADE agent environment [22]. Finally, it is planned the application of the previously tested JADE-based MAS for control purposes of a real bioreactor.

Acknowledgments. This work was supported by the Polish Ministry of Science and Higher Education under grant no. N N514 471539 and grant BK-UiUA.

References

1. Wooldridge, M., Jennings, N.R.: Intelligent agents: theory and practice. Knowl. Eng. Rev. 10, 115–152 (1995)
2. Jennings, N.R., Sycara, K., Wooldridge, M.: A Roadmap of Agent Research and Development. Auton. Agent. Multi-Ag. 1, 7–38 (1998)

3. Wooldridge, M. (ed.): An Introduction to Multiagent Systems. John Wiley & Sons, New York (2002)
4. Nguyen, N.T., Katarzyniak, R.: Actions and social interactions in multi-agent systems. Knowl. Inf. Syst. 18, 133–136 (2009)
5. Marik, V., McFarlane, D.: Industrial Adoption of Agent-Based Technologies. IEEE Intel. Syst. 20, 27–35 (2005)
6. Pechoucek, M., Marik, V.: Industrial deployment of multi-agent technologies: review and selected case studies. Auton. Agent. Multi-Ag. 17, 397–431 (2008)
7. Van Dyke Parunak, H.: A practitioners' review of industrial agent applications. Auton. Agent. Multi-Ag. 3, 389–407 (2000)
8. Metzger, M., Polakow, G.: A Survey on Applications of Agent Technology in Industrial Process Control. IEEE T. Ind. Inform. 7, 570–581 (2011)
9. Nocon, W., Metzger, M.: Predictive Control of Decantation in Batch Sedimentation Process. AICHE J. 56, 3279–3283 (2010)
10. Skupin, P., Metzger, M.: Cooperative Operating Control for Induction or Elimination of Self-sustained Oscillations in CSTB. In: Luo, Y. (ed.) CDVE 2011. LNCS, vol. 6874, pp. 66–73. Springer, Heidelberg (2011)
11. Choinski, D., Metzger, M., Nocon, W.: MAS-Based Cooperative Control for Biotechnological Process-A Case Study. In: Mařík, V., Strasser, T., Zoitl, A. (eds.) HoloMAS 2009. LNCS, vol. 5696, pp. 175–182. Springer, Heidelberg (2009)
12. Chen, C.I., McDonald, K.A., Bisson, L.: Oscillatory behavior of Saccharomyces cerevisiae in continuous culture: I. Effects of pH and nitrogen levels. Biotechnol. Bioeng. 36, 19–27 (1990)
13. Satroutdinov, A.D., Kuriyama, H., Kobayashi, H.: Oscillatory metabolism of Saccharomyces cerevisiae in continuous culture. FEMS Microbiol. Lett. 98, 261–268 (1992)
14. Parulekar, S.J., Semones, G.B., Rolf, M.J., Lievense, J.C., Lim, H.C.: Induction and elimination of oscillations in continuous cultures of Saccharomyces Cerevisiae. Biotechnol. Bioeng. 28, 700–710 (1986)
15. Balakrishnan, A., Yang, R.Y.K.: Self-forcing of a chemostat with self-sustained oscillations for productivity enhancement. Chem. Eng. Commun. 189, 1569–1585 (2002)
16. Bader, F.G.: Analysis of double-substrate limited growth. Biotechnol. Bioeng. 20, 183–202 (1978)
17. Metzger, M.: Mathematical model for continuous culture under double substrate limitation with the specific growth rates proportional to the ratio of flow rates. Internal report of the Institute of Automatic Control, 1–12 (2009)
18. Skupin, P.: Simulation approach for detection of the self-sustained oscillations in continuous culture. In: Proceeding of the 11th WSEAS International Conference on Mathematics and Computers in Biology and Chemistry, Iasi, pp. 80–85 (2010)
19. Metzger, M.: A comparative evaluation of DRE integration algorithms for real-time simulation of biologically activated sludge process. Sim. Pract. Theory 7, 629–643 (2000)
20. Shamma, J. (ed.): Cooperative Control of Distributed Multi-Agent Systems. John Wiley & Sons, New York (2008)
21. Choinski, D., Senik, M.: Multi-Agent oriented integration in Distributed Control System. In: O'Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS (LNAI), vol. 6682, pp. 231–240. Springer, Heidelberg (2011)
22. Bellifemine, F., Caire, G., Greenwood, D. (eds.): Developing Multi-Agent Systems with JADE. John Wiley & Sons, Chichester (2007)

Introducing Session on ICT-Based Alternative and Augmentative Communication

Željka Car[1], Dinka Vuković[2], Nadica Bjelčić[3], Goran Karas[4], and Velimir Karas[4]

[1] University of Zagreb, Croatia
Zeljka.Car@fer.hr
[2] Croatian Down Syndrome Association
dinka.vukovic@gmail.com
[3] OKO Association of Parents of Visually Impaired and Multi Disabilities Children
nadicab@gmail.com
[4] AmoReVera - Association for Persons with Intellectual and Communication Impairments
amorevera.hr@gmail.com

Persons with complex communication needs are individuals who are unable to use speech as their primary method of communication. Reasons for complex communication needs differ widely, such as physical and/or intellectual impairment, Down syndrome, Autistic Spectrum Disease etc. Alternative and augmentative communication (AAC) is based on the use of graphic and textual symbols to represent certain objects, actions, or concepts. AAC has been proven as highly beneficial for improving speech, literacy, learning, employment, and quality of life for persons with complex communication needs. ICT-based AAC services refer to services that enable symbol-based human-to-human and human-to-machine communication in a computer and networking environment.

Alternate and augmentative communication offers children with developmental delays an instant preview of typical communication situations; it encourages expressive and receptive language development and serves as a bridge to future speech development. Children who used *Picture Communication Symbols* (PCS) were more efficient in speech.

Based on research, as well as on the basis of parental experience in the independent use of AAC, it is recommended to apply AAC as the first means of fostering the development of speech, where a child is provided information via three different channels: speech (auditory) + image (visually) + gesture (body language).

Children with Down syndrome are ready to communicate much earlier than they speak and they better understand language than they can use it. The disproportion between understanding and speaking a language at an early age can be two to three years and in some cases even more. For that reason it is necessary to develop language understanding in parallel with the development of alternative and augmentative communication skills.

Communications skills of children with multiple impairments, including serious motor disabilities and/or vision impairment, lag behind the skills of their peers. It is also very difficult to introduce the sign language because of the motor problems. At an early age the stress is usually placed on improving a child's

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 219–220, 2012.

motor abilities, so other areas such as sensor, communication, oral skills and fine facial motor skills are neglected. In most cases parents seek support of speech therapist when their child is at the age of three.

Unfortunately, parents still have to navigate on their own through the system of communication and education habilitation. Experts who know how to initiate communication with children with motor impairments are few, especially where children of multiple impairments, including vision impairment, are concerned. There is no systematic plan to develop communication skills, i.e. basic terminology to start mastering communication has not been devised.

Persons with intellectual and communication impairments are considered as one of the most socially isolated people in the society, mainly caused by the strong presence of stigmatization which sets certain rules and behaviour patterns. In the last 50 years, they have gained many legal rights that improved various aspects of their lives. However, these rights cannot ensure acceptance from their peers, neighbours, colleagues or any other group which directly affects the lives of persons with the impairments.

So far, persons with intellectual and communication impairments haven't got a real chance to have friends out of their special educational environment. We are witnessing that deinstitutionalization and implementation of the ICT based augmentative and alternative communication solutions is improving these chances thus removing the barriers that separate them from other parts of the society.

Involving the parents in the research and development of ICT-based AAC service aims at better identification of research problems, efficient requirements definition and obtaining experienced quality feedback for service evolution and improvement. Recommendations for development of interface for AAC applications based on parents experience are the following: Alternative communication should be based on big images without distracting details. Secondary motives should be avoided. It is advisable to use Arial font because of its simplicity and visual clarity. Red colour is the quickest to reach the brain due to its frequency so it can be used to emphasize individual letters or words.

Parental activities go beyond recommendations: a family with the 22 year old son with autism and facing the lack of IT services in Croatian was encouraged to develop Croatian AAC communicator, including AAC portal with the Web-Communicator as its essential part. The name of the portal is an abbreviation for "*I communicate*" in Croatian language, which puts emphasis on its primary purpose.

Collaboration efforts of multidisciplinary researchers, software developers and parental associations is crucial for designing interoperable and scalable AAC services by using smart information and communication technologies including software agents.

Five papers within this invited session cover original submissions on technical and non-technical aspects of developing and using ICT-based AAC services and investigate potential of applying tablet devices and tangible user interfaces in this field. The overall goal of all undertaken researches is to advance social inclusion of people with complex communication needs by using teamwork synergy of experts from different fields and grasping new technology potential.

The Use of AAC with Young Children in Croatia– from the Speech and Language Pathologist's View

Jasmina Ivšac Pavliša[1], Marta Ljubešić[1], and Ivana Jerečić[2]

[1] University of Zagreb, Faculty of Education and Rehabilitation Sciences, Croatia
jivsac@erf.hr, marta.ljubesic@public.carnet.hr
[2] Suvag Polyclinics, Croatia
meritamonlog@gmail.com

Abstract. All children with complex communication needs have the right to engage in interpersonal communication and participate in society. The field of augmentative and alternative communication (AAC) provides education, systems and practice that enable such participation. AAC requires a multidisciplinary approach. A speech and language pathologist (SLP) is an important member of the team responsible for assessment, implementation and evaluation within AAC. This paper presents the current knowledge and practice of SLPs in Croatia in the area of AAC. The data shows that SLPs in Croatia rarely use aided and high-technology tools with clients. Three cases of children with low levels of expressive communication are elaborated from the aspect of their needs and intervention. It is concluded that the area of AAC has a great need for attention in the following: 1) further promotion among the general public, 2) multidisciplinary collaboration and 3) additional education for professionals.

Keywords: early intervention, interpersonal communication, practice, level of awareness of AAC.

1 Introduction

The concept of augmentative and alternative communication began to develop back in the 1950s in order to augment or provide an appropriate mode of communication for people who could not communicate in a conventional way [1]. Nowadays, people who cannot communicate in conventional ways due to reasons such as speech, language, motor and/or intellectual impairments are known as people with complex communication needs [2]. Although the field of augmentative and alternative communication (AAC) has been highly developed in terms of strategies and technologies, its implementation and use is nowadays qualitatively different in different societies. Within Croatia, the general public has a low level of awareness of the concept of AAC. In clinical practice, more and more speech and language pathologists as well as other professionals have been expressing an interest in its development but the systematic use of AAC is still an exception rather than the rule.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 221–230, 2012.

The results of 101 questionnaires completed by speech and language pathologists in Croatia show that most speech and language pathologists use AAC in their work (67.33%), and most of those who do not use it have seen its use in practice [3]. Furthermore, the analysis of the AAC systems shows gestures are the dominantly used system (70.59 %). Paper and pen are also used to a great extent (67.65%) as well as the Picture Exchange System (48.53%) and sign language (32.29%). Various computer software is not used as frequently (27.94%) as gestures and signs. Other low technology systems are used rarely: communication books 8.82% and communicators 5.88%. The gathered data clearly shows that unaided methods are more dominantly used among Croatian speech and language pathologists than aided and high-technology tools, which are used rather rarely (Figure 1).

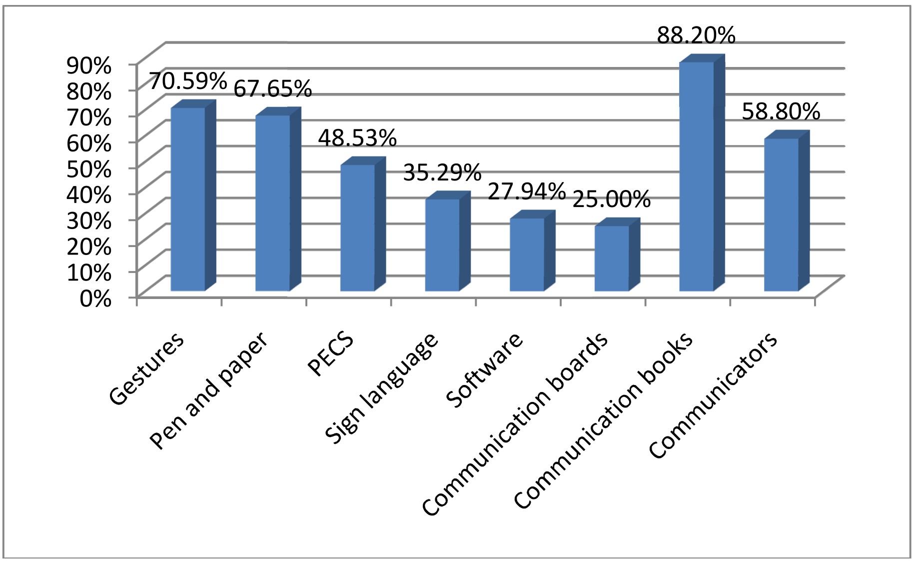


Fig. 1. The distribution of different AAC systems that speech and language pathologists in Croatia use in their work with clients [3]

A qualitative analysis of the gathered data and comments leads to the conclusion that many speech and language pathologists do not have an appropriate concept of AAC. Namely, they stated which systems they use in their everyday work (paper, pen, various software) but do not necessarily use these systems for the purposes of the AAC. This fact addresses the need for further training of those speech and language pathologists who completed their professional education years ago. Furthermore, the AAC course has been a regular part of the speech and language pathology study program for the past three years (www.erf.hr). It can be emphasized that aided tools are rarely used among speech and language pathologists in Croatia. It is possible that the use of high-technology tools is also a result of a low level of cooperation between speech and language pathologists and technical professions. In order to improve the field ofthe AAC in Croatia, the grounds for a multidisciplinary approach have been recently established [4].

On the other hand, the parents and people who are in contact with young children still have reserves about the use of AAC and tend to lean on famous myths about its use [5]. Clinical experiences gathered over many years in the Early Communication Unit have shown that people usually worry that access to AAC will hinder or stop the speech development of the child [6]. All the listed facts point to the great need for the promotion and development of the field of AAC in Croatia among the general public as well as among professionals, through interdisciplinary collaboration. The case studies presented in this paper will also depict the state of practice in Croatia where professionals unfortunately do not start with AAC methods early enough.

1.1 Specific Disabilities and AAC

It is well known that the needs of people with complex communication needs are highly individualised and complex and that the term itself refers to a wide group of disabilities and disorders [7]. There are numerous classifications trying to categorise the users of AAC methods [8, 9, 10]. The most common developmental disorders related to complex communication needs are autism and autism spectrum disorders, cerebral palsy, intellectual disabilities and childhood apraxia of speech [9]. The categorisation that uses the levels of language comprehension and expressive language as a criteria divides AAC users into three groups [8]: (1) the expressive language group (problems with speech, appropriate level of language comprehension), (2) the supportive language group (children who temporarily use AAC as a way of facilitating language comprehension as well as to express themselves and those children whose speech is unintelligible) and (3) the alternative language group for which AAC is a permanent means of communication. The basic roles are also defined using the temporality and severity of complex communication needs: augmenting existing natural speech, providing a primary output mode for communication, providing an input and an output mode for language and communication and serving as a language intervention strategy [5].

The majority of AAC publications have been focused on adults with complex communication needs [11]. But the results of contemporary research point to the relevance of implementing the AAC system in early age [11, 12]. The role of AAC, as an early intervention method, is also defined by the child's needs. The early implementation of AAC methods promotes the communication process for the whole family. Namely, AAC makes the child' s needs and desires clear, enabling communicative partners to appropriately react to them. All this facilitates communication and language development. Longitudinal studies focused on the outcome of AAC intervention in young children with significant communication disabilities (16-36 months) showed that all children demonstrated improvement in crucial communication skills: rate of turn taking, sustaining interactions longer, participation in routines, play activities [11, 13]. Furthermore, all children have acquired a range of semantic concepts and showed improvement in terms of a greater complexity of language structure. The potential for implementing AAC at an early age has not been fully realized for young children [13].

1.2 Key Points about Communication and Language

AAC is a multimodal intervention approach with an aim of improving the communication competence of the person with complex communication needs [2]. In order to fully comprehend the communicative needs of AAC users and the approach itself it is essential to gain insight into some key features of communication and language development.

The great majority of children raised in a relatively normal language-using community will, without formal education, develop the ability of using that community's language. There are numerous definitions of language, but most of them emphasize the fact that language is embedded in communication [6]. Language development is a highly complex and dynamic process which is fundamentally grounded in different sets of important components that can be considered prerequisites for language development (reciprocal behaviours, intentionality in communication, joint attention, canonical babbling, etc.). All of these can be seen in prelinguistic development, but children also may have deficits in some of the prerequisites which preventing them from developing speech and language. As many children with complex communication needs remain in the phase of prelinguistic development it is important to start with AAC intervention as early as possible and understand that the same phase can mean an entire set of different features and abilities for each child. Long before children are able to talk they become aware that they can influence other people through other means of communication (between 6-12 months in a typical development - gazing, vocalizing, gesturing). This moment in development is called the start of intentional communication. Many children who are candidates for AAC have not reached this level, and their communicative partners (parents or teachers) interpret their needs based on their behaviour, which means that many AAC candidates stay at the preintentional level. Entering the level of intentional communication enables the child to start involving others in his/her interests and sharing them. This is so-called triadic communication (you and I communicate about something). It is well known that triadic communication is one of the prerequisites for the discovering of the meaning of words and sentences.

After the child has reached the level of triadic communication we have to examine the communicative pattern the child has. The communication pattern is defined through communicative functions (the reasons why the child communicates), which are extremely important in the field of AAC because we have to provide the child with symbols he or she needs in order to realize his communicative purposes. In typical development children communicate for a variety of purposes: to comment on something, to demand or ask for things or objects, to report etc. People with complex communication needs usually communicate for a limited range of communicative functions. For this reason, one of the key points in AAC intervention is to teach the child how to enlarge the range of communicative purposes. This means that the speech and language pathologist has to identify the purposes the child communicates for and to teach him how to use symbols for new purposes -for example, to ask for help, to ask for a specific object, to demand or refuse a certain kind of food etc.[14]. In the prelinguistic stage, understanding language is mainly situational, which means that the child shows appropriate reactions to verbal directions only in highly routinized and everyday situations. This is the reason why intervention also includes strategies to improve language understanding.

2 Case Studies – Assessment, Communicative Needs and AAC as an Intervention Method

Through three case studies we would like to depict the need for AAC from a group of AAC users with developmental disabilities. The following examples will describe a child with verbal dyspraxia, a child with profound intellectual disabilities and a child with complex motor impairments and communication disabilities. Their only common feature is a low level of expressive communication but based on the results of the assessment it will be clear that beneath this problem there is a whole array of different abilities and deficits. All the children's communication abilities and needs were assessed in the Early Communication Unit[1]. According to the results of the assessment the team members decided about the AAC method for each child individually.

2.1 A Child with Profound Intellectual Disabilities (Age 4.9)

A girl at high neurodevelopmental risk (ex. born in the 25th week of gestation, PW/PL 973 g/47 cm, APGAR 4/5) and with proven perinatal brain damage came for professional assessment at the age of 4.9. The parents emphasized their concern about a delay in the speech development.

From the cognitive aspect, the girl has a very limited attention span and difficulties in sharing attention with others. She does not understand the communicative function of the words she hears. Her performance on cognitive tests points to a moderate delay in cognitive development.

An assessment of communication and language pointed to significant problems in the use of communicative functions and means. At the age of 4.9 the girl communicates dominantly for imperative purposes and uses presymbolic means of communication (contact gesture and vocalisations). Spontaneous vocalisations as well as syllables (without any meaning) are very rare. The girl uses no words.

This girl has been continuously included in different therapy procedures throughout her life. In spite of significant delays in communication development and no emergence of speech, even in this era of high technology none of the AAC methods had yet been introduced in working with the girl.

As all the mentioned aspects of communication and language understanding showed to be significantly delayed the professionals from the Early Communication Unit (a psychologist and speech and language pathologist) agreed that this girl needs a

[1] The Early Communication Unit is specialised in the speech and language and psychological assessment of young children with different developmental disorders. It is one of the units of Centre for Rehabilitation of the Faculty of Education and Rehabilitation Sciences.

different input and output mode for communication. After the assessment the girl was included in a home support programme designed so that a speech and language pathologist with clinical experience with young patients made home visits. Methods of AAC have been introduced in the child's natural setting. This girl is in the stage of so-called emerging communication with a dominance of non-symbolic methods of communication [15]. As the girl does not understand any language, as she has low imitation skills and is moderately delayed in the cognitive development, the use of gestures or complex communicators could not be tools of choice for her. In this case, the intervention is structured with an aim to provide an input and an output mode for language and communication. The first goals are to establish some behaviour that stand for/represent an action, person or an object. As the first choice, the speech and language pathologist tried to implement the Picture Exchange System - PECS [14, 16]. Communicative episodes using PECS have been infused in all activities (at home, in the preschool setting, in the park, at the grandmother's house) and with different communication partners (eg. peers, relatives, etc.). The whole intervention should be built within the child´s meaningful contexts starting with pictures that are familiar and highly motivating to the child. In the first steps all efforts were focused on engaging the girl in functional communication so that she tries to ask for her favourite toy or other object and then to expand it to other communicative functions (ex. asking for more, asking to stop, responding, commenting.). The long-term goal is for the girl to start using symbols for small conversations (ex. where she has been, etc.) and expanding the AAC tools (ex. talking photo album, iPad). This girl has tried using iPad and it was easy for her to navigate through, but until now there was no appropriate application for her to use. It is expected that collaboration with ICT professionals will result in a communicator application for the iPad which will enable a choice of those symbols which are important for the girl´s communicative needs (f.e. asking for a preferred toy).

Although her parents were exclusively fixated on the development of speech, they eventually accepted pictures as a mode that would help the whole family interact more successfully. This case study relates to one of the myths about the use of AAC - the myth that children must have a certain set of skills to be able to benefit from AAC [5]. In spite of her low level of cognitive and communication skills, this girl will surely benefit from the AAC.

2.2 A Child with Developmental Apraxia (Age 5.1)

A boy with clear anamnestic data (regular second pregnancy and labour) came for an assessment at the age of 5.1. His motor development was typical. According to the parents, his speech and language development was slower from the very beginning (delayed emergence of first words).

The boy showed appropriate social interests. Cognitive assessment pointed to milder delays in different performance tasks.

Language assessment revealed lexical understanding and understanding of simpler demands (Reynell Developmental Language Scales). The level of language comprehension is 3.3, showing a significant delay. In spontaneous communication the child uses only 10 words of simple structure and vocalisations. There is no lexical imitation. Results of the assessment pointed to a discrepancy between the child's expressive and receptive language. The boy has rich facial expressions and his own way of exchanging messages with other (ex. as a sign for blue colour he will point to the sky). The boy has showed a typical communicative pattern and good cognitive potential but also a significant delay in expressive communication and a mild delay in the segment of language comprehension. This huge discrepancy between understanding and expression pointed to childhood apraxia of speech.

Although the parents searched for professional help on several occasions, the boy has never been included in speech and language therapy and there were no attempts to introduce AAC.

This boy's numerous communicative needs are unsatisfied due to an obvious lack of communicative means and therefore his communication within unfamiliar environments is very restricted. This level of communication is considered context-dependent because his communication is clear with very familiar partners and in specific contexts [15]. As the boy shows milder delays in the cognitive domain and as his strong communicative feature is language understanding and appropriate social interest he firstly needs a primary output mode for communication -gestures, pictures or a more sophisticated system of communication which will enable him to communicate beyond yes or no responses. The use of aided and/or unaided symbols will enable participation in conversation with different communicative partners (ex. sharing information, asking, reporting, etc.). As he spontaneously found a way to exchange messages with others through simple gestures and as he has good imitation skills, the use of gestures should be a good starting point in the implementation of AAC. Apart from expanding a number of communicative functions and participation among peers, the implementation of AAC should contribute to the development of semantic concepts and complexity of language structure (morphosyntax, vocabulary). In this case, the iPad would also be a tool of choice, but this software application should encompass more complex possibilities of choices. As the boy understands language quite well, the application should enable constructing simple sentences and questions (ex. I want to go for a walk, Where is the mother?).

This boy could not be offered home visits and immediate help because the family lives too far from the capital of Croatia which, unfortunately, further limits the chances for appropriate professional support.

2.3 A Boy with Severe Motor Impairments (Age 2.02)

A boy is born from mother's first pregnancy complicated with several urinal infections. He was born in the 37th week of gestation, APGAR 10/10. In the seventh month of life a regression in motor development was noticed. Ultrasound findings pointed to periventricular echogeny. Genetic analysis showed a suspicion of spinal muscular atrophy. The boy does not walk at the age of 2.02.

The boy is included in physical and occupational therapy and a home support programme.

Results on the Bayley Scales of Infant and Toddler Development revealed a significant delay. The skills needed for everyday living are in line with the functioning of a nine-month-old child (Vineland Adaptive Behaviour Scales). This result is a consequence of motor impairments; it is not due to difficulties in cognitive or social development.

His communication is at a pre-intentional level (parents guess about his desires and feelings based on his behaviour). Joint attention skills were not developed (he fixates the object of desire, without including communicative partners). The level of language understanding could not be precisely defined: motor response is unclear. Parents report that the boy uses some words (as associations) but without any communication function (to ask for something).

The assessment revealed an uneven developmental profile with a significant delay in the segment of motor development and communication. His motor impairments influence his whole learning process and his ability to respond to stimuli from his environment.

As this boy is already included in a home support programme (age 2.02), this is a positive example of recent changes in the area of early intervention in Croatia. Due to severe motor problems the use of his hands is very restricted (ex. he cannot pick up a picture in order to exchange it for an object). A first step of the early intervention programme is to strengthen the family in their everyday routine and coping with the child and facilitation of the boy's communication skills. As he is at the pre-intentional level, as he shows a significant delay in the cognitive domain and has severe motor problems, the implementation of AAC methods is crucial in his case [17]. Due to low levels of the imitation skills and language understanding, the talking album or Speech Generating Device are the starting points in the intervention procedure. Professional support at home began with the use of the talking photo album because the child was able to start the album himself. The first attempts aimed to teach him to ask for his favourite food by pressing a switch on the album, and in that way to facilitate the emergence of intentional communication. The parents welcomed this decision and helped the professional choose the photographs and started to infuse this way of asking for objects and activities in many everyday situations. Once the child gains an understanding that he can reach his parents or other communicative partners through pictures, a tablet with the communicator application will be introduced. The communicator application should enable a choice of different symbols (ex. food, activities, toys) that this boy prefers. Due to the problems with motor coordination, this boy will need special AAC requirements in the form of adjusted switches which start the application on the iPad and/or less sensitive touch screen. The Speech Generating Device will also be used (ex. Bigmack) until the same application for the iPad is developed. The applications used with this boy will, hopefully, be a result of multidisciplinary collaboration referred to in the beginning of this paper [4].

3 Conclusion

Although the results of the research regarding the implementation of AAC methods by the speech and language pathologists [3] and the description of the first two case studies (rather late onset with AAC) is not very promising, some positive steps have been made in several areas. Firstly, the formal education of future speech and language pathologists has improved over the last three years with the introduction of a course on AAC in their studies. Secondly, the specialised postgraduate study programme in *Early Intervention* that educates specialists in early intervention offers broad and multidisciplinary knowledge in the segment of early years and in the concept of AAC. As a direct outcome of improvements in training there will gradually be more and more competent speech and language pathologists who will understand the importance of starting early and how to support interpersonal communication with young children with complex communication needs [18]. Hopefully, their work will be supported by the results of the multidisciplinary collaboration that has recently been initiated and that has as a focus the adaptation of high technology tools for children with complex communication needs [4]. The children with complex communication needs are different so the software applications and the devices should be adjusted to their individual needs (ex. the content, situations and the people engaged in communication) and the features if each child. It is expected that the mentioned collaboration will enable the creation of the software applications that could be adjusted to the child´s physical, perceptive and cognitive abilities.

Acknowledgements. This work was carried out within development project "ICT systems for Persons with Complex Communication Needs", supported by University of Zagreb Development Fund.

References

1. Hourcade, J., Everhart Pilotte, T., West, E., Parette, P.: A History of Augmentative and Alternative Communication for Individuals with Severe and Profound Disabilities. Focus on Autism and Other Developmental Disabilities 19(4), 235–244 (2004)
2. King, J.M.: Complex Communication Needs and AAC. In: Justice, L.M. (ed.) Communication Sciences and Disorders: An Introduction, pp. 501–529. Pearson Merrill Prentice Hall (2006)
3. Jerečić, I.: Uporaba potpomognute komunikacije u logopedskoj praksi. Diplomski rad. Edukacijsko-rehabilitacijski fakultet. Sveučilište u Zagrebu (2011)
4. Stančić, Z., Frey Škrinjar, J., Ljubešić, M., Car, Ž.: Multidisciplinary Collaboration and ICT Services for People with Complex Communication Needs. In: Biljanović, P., Skala, K. (eds.) MIPRO 2011 Proceedings of the Conferences Microelectronics, Electronics and Electronic Technology and Grid and Visualization Systems, pp. 265–270. MIPRO, Rijeka (2011)
5. Romski, M.A., Sevcik, R.A.: Early Intervention and Augmentative Communication: Myths and Realities. Infants and Young Children 18, 174–185 (2005)
6. Ljubešić, M., Ivšac Pavliša, J., Cepanec, M., Šimleša, S.: What the Features of Prelinguistic Communication can tell us about Developmental Trajectories. In: Gačnik, M., Strle, M. (eds.) Conference Proceedings and Posters and Workshops Abstracts 3rd Congress of Slovenian Logopedists with International Participation, pp. 80–85. Centar za korekciju sluha in govora, Aktiv logopedov CKSG, Portorož (2011)

7. Goldbart J., Caton, S.: Communication and People with Most Complex Needs: what works and why this is essential, Research Institute for Health and Social Change. The voice of learning disability. Mencap (2010)
8. von Tetzchner, S., Martinsen, H.: Introduction to Symbolic and Augmentative Communication. Singular, San Diego (1992)
9. Beukelman, D.R., Miranda, P.: Augmentative and Alternative Communication: Management of severe communication disorders in children and adults. Paul H. Brookes Publishing Company, Baltimore (2005)
10. Murray, J., Goldbart, J.: Augmentative and Alternative Communication: a review of current issues. Paediatrics and Child Health 19(10), 464–468 (2010)
11. Branson, D., Demchak, M.: The Use of Augmentative and Alternative Communication Methods with Infants and Toddlers with Disabilities: A Research Review. Augmentative and Alternative Communication 25(4), 274–286 (2009)
12. Cress, C.J., Marvin, C.A.: Common Questions about AAC Services in Early Intervention. Augmentative and Alternative Communication 19(4), 254–272 (2003)
13. Light, J., Drager, K.: AAC technologies for Young Children with Complex Communication Needs: State of the science and future research directions. Augmentative and Alternative Communication 23, 204–216 (2007)
14. American Speech-Language-Hearing Association. Roles and Responsibilities of Speech-Language Pathologists With Respect to Augmentative and Alternative Communication: Position Statement (Position Statement) (2005), http://www.asha.org/policy
15. Blackstone, S.W., Hunt Berg, M.: Social Networks: A Communication Inventory for Individuals with Complex Communication Needs and their Communication Partners. Augmentative Communication, Inc., Monterey (2003)
16. Charlop-Christy, M., Jones, C.: The Picture Exchange Communication System. In: McCauley, R., Fey, M. (eds.) Treatment of Language Disorders in Children, pp. 105–122. Paul H. Brookes, Baltimore (2006)
17. Reichle, J., Halle, J.W., Drasgow, E.: Implementing Augmentative Communication Systems. In: Wetherby, A.M., Warren, S.F., Reichle, J. (eds.) Transitions in Prelinguistic Communication, pp. 417–437. Paul H. Brookes Publishing (1998)
18. American Speech-Language-Hearing Association. Principles for Speech-Language Pathologists in Diagnosis, Assessment, and Treatment of Autism Spectrum Disorders Across the Life Span (Technical Report) (2006), http://www.asha.org/policy

AAC Services Development: From Usability Requirements to the Reusable Components

Ivan Vučak[1], Hrvoje Belani[2], and Marin Vuković[3]

[1] Sedam IT, Koledovčina 2, HR-10000 Zagreb, Croatia
[2] Croatian Institute for Health Insurance, Margaretska 3, HR-10000 Zagreb, Croatia
[3] University of Zagreb, Faculty of Electrical Engineering and Computing, Department of Telecommunications, Unska 3, HR-10000 Zagreb, Croatia
ivan.vucak@sedamit.hr, hrvoje.belani@hzzo-net.hr, marin.vukovic@fer.hr

Abstract. Intensive development of information and communication technologies can contribute greatly to enhance alternative and augmentative communication for individuals who experience difficulty communicating in an understandable manner. Availability of internet infrastructure and affordability of mobile computer devices provide a sound starting point for planning and building AAC services. In order to make the services more familiar and understandable to users, it is justifiable to build them on top of common service platforms in an adaptive and configurable manner. In this paper we present a component-based AAC service development model that enables building such services by using mandatory AAC components on top of the component framework, and selecting visual or non-visual components from the available component pool. The AAC component selection criteria are based on matching the data and metadata of usability requirements specified with the metadata of available components from the component pool. Registering new components into the reusable component pool extends the pool and helps building the component-based AAC services development platform.

Keywords: alternative and augmentative communication, component-based development, usability requirements, component pool, AAC services.

1 Introduction

The means of communication in today's interconnected world are more diverse than ever. Continuous growth, availability and affordability of information and communication technologies (ICTs) are considered crucial for bringing more people into information age [1], simultaneously enabling new ways of information exchange and content production, offering new and transformative applications and services [2]. Raising popularity and affordability of mobile computers, like tablets and smartphones, along with the common mobile phones and notebook PCs, provides an access capability for various growing and emerging user groups to different ICT services. Nevertheless, in order for these services to be fully accepted and utilized by the users, along with the proper functionality they have to offer, their priory need is to be usable and trustworthy.

In order to make the services more familiar and understandable to users, it is justifiable to build them in an adaptive and configurable manner, therefore relying on software reuse.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 231–240, 2012.

This is exactly one of the characteristics of component-based software engineering (CBSE), which aims at creating software systems and ICT services faster at lower costs and with increased stability. A software component is a small independent unit of composition with contractually specified interfaces and explicit context dependencies only. It can be deployed independently and is a subject to composition by third parties [15]. In order to build a component-based system, as any other software system, the corresponding requirements specification has to be fulfilled, i.e. the requirements implemented. There are various requirement categories and types, along with their interdisciplinary nature, which brings complexity into narrowing the requirements-implementation gap. This is the main reason why we limit the requirements scope in the paper to usability requirements, which are of crucial importance to implementation of ICT services for enhancing alternative and augmentative communication (AAC).

This paper is organized as follows: the next section elaborates requirements engineering for AAC services development. The third section provides an approach of component-based software development to AAC field, proposing an AAC component development framework. The fourth section analyses the non-functional requirements and architectural components interdependency inside the framework, providing an AAC service example. The last section gives conclusion, along with the comments on platform-based AAC service development and future work plans.

2 Requirements Engineering for AAC

Both for ICT services and the underlying devices themselves, usability can be defined as an extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use [3]. Applying one of the known trust models to ICT services, we may consider them trustworthy if they: (1) have ability to fulfill requests in a given domain, (2) have integrity to provide proper results as stated, and (3) are benevolent – act in user's best interest [4]. However, the scope of this paper is not trustworthiness, but dealing with usability-related requirements and component-based development of AAC services.

Augmentative and alternative communication, or vice versa (alternative and augmentative communication), is “the supplementation or replacement of natural speech and/or writing using aided and/or unaided symbols” where “use of aided symbols requires a transmission device” [5]. It is a field of study concerned with providing devices and techniques which aim to augment the communicative abilities of individuals who experience difficulty speaking, writing and other skills of communication in an understandable manner [6]. As stated in [7], the main goal and purpose of all assistive technology for AAC is to provide individual user with the ability to generate meaningful phrases and express his/her feelings, thoughts, needs and desires. Depending on the individual’s skills and abilities, the decision has to be made which device to use, among various devices that contain different features and different levels of complexity.

Applying the same statements to ICT services for enhancing AAC, it is evident that various services have to be offered, in order to fulfill different user requirements and needs. However, having a great number of various services on different platforms, without an efficient way of precisely choosing the desired service features and properties, is not contributing effectively to the targeted user group. In this paper we state the need for introducing a component-based model for developing AAC services in a configurable and adaptive manner. It involves building and updating the library of reusable AAC components, both visual and non-visual, which can be chosen for building a new AAC service regarding the specified usability requirements.

2.1 Usability Requirements Taxonomy

The requirements for a system can be described as descriptions of the services provided by the system and its operational constraints [8]. They reflect the needs of customers for a system that helps solve some problem or answer on demand, such as controlling a device, placing an order of finding information. In a software industry over several decades, requirements were considered as labels for abstractions used to carry the value stream into development and on to delivery to customers [9]. Practically all existing requirements engineering (RE) paradigms put customers in central place, but most of the mainstream RE methods and techniques, such as surveying [10] and interviewing, are not completely suitable for immediate users in our case, the persons with disabilities and special needs. Regardless the usage of traditional or agile RE approaches [9]; we recommend these techniques to be carefully adopted for the customers and users that are not typically considered during the ICT service design process. Also, techniques should actively include people who support these users, such as caretakers and guardians. Diversity of user population should be analyzed, as well as a wide spectrum of functional (FR) and non-functional requirements (NFR) that should be implemented under the same service.

For the purpose of this paper, we have adopted the usability requirements taxonomy to serve as a starting point for requirements specification in our AAC service development model. In a broader sense, usability requirements, given in Table 1, should describe the means of perceived usefulness and ease of use for the specified user interface, and the service as a whole.

Table 1. Types of usability (UR) requirements (adopted from [11])

No.	*Requirement name*	*Requirement description*
URT1	Accessibility	Ease of access to and use of specific functionality.
URT2	Aesthetics	Aesthetics of the user interface and description of "look and feel".
URT3	UI consistency	Consistency of the user interface, both within the system and with other systems.
URT4	Ergonomics	Aspects of the user interface, such as avoiding unnecessary clicks, uncomfortable touchscreen moves, etc.
URT5	Ease of use	Ease of learning and using the system.

2.2 Requirements Interchange Format

While natural language requirements themselves are too demanding to be managed efficiently and their implementation status traced throughout the development process, it is advisable to provide a tool with an appropriate, machine-processable way of specifying and storing them. For the purpose of this paper we have chosen to specify requirements in so-called Requirements Interchange Format (ReqIF) by using the open-source tool ProR [12].

ProR is Eclipse-based tool for requirements engineering that supports the emerging RIF/ReqIF standard [13] which provides interoperability with other authoring requirements management tools in industry. ReqIF is an XML-based file format used to exchange requirements, along with its associated metadata, in an open, non-proprietary and tool-independent manner. One of the main ReqIF features is hierarchically structured specification of uniquely identified requirements that have associated attributes to them and established relations between them. Typical ReqIF file also contains data type definitions and tool extensions information, in order to maintain interoperability between tools. The main vision of ProR is to provide reliable traceability between natural language requirements (NLRs) and formal models [12], namely in Event-B notation [14].

2.3 AAC Service Example

To illustrate AAC service usability requirements, here we describe the basic communicator service for people with special needs. These are few examples of NLRs for a given AAC service, regarding the proposed requirement types:

- Accessibility – *"Access to the library of symbols shall be available from the main screen"*,
- Aesthetics – *"All buttons shown on the main screen should be vertically aligned to the right side"*,
- UI consistency – *"User interface shall be consistent with W3C WAI AA standard [http://www.w3.org/WAI/WCAG1AA-Conformance]"*,
- Ergonomics – *"When a symbol-based dialog box is opened, the focus shall be on the first entry field in the dialog box"*,
- Ease of use – *"The application has to offer a HELP menu with instructions for users"*.

After configuring (labels, data types, requirement types, etc.), customizing and using the ProR tool to gather and specify service requirements into the ReqIF project, the example of usability requirements specification for a given AAC service is shown on Figure 1. Besides the description, every requirement has a certain number of attributes that specify it in more details, e.g. requirements source, creation date, owner and status. The given specification can be exported as the corresponding ReqIF file, which serves as an input for a component-based model. The data and the metadata from the ReqIF file can be matched with the metadata describing available AAC service components, helping developers decide to include it into the service or not.

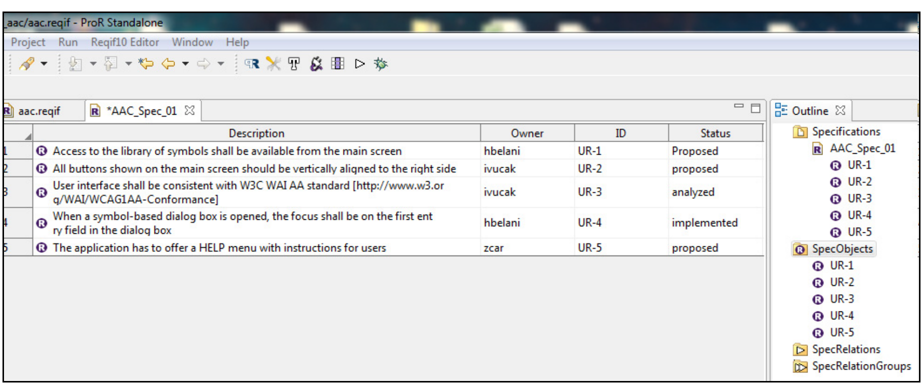


Fig. 1. AAC service usability requirements specification in ProR tool

3 Component-Based Software Engineering

CBSE emerged from the failure of object-oriented development to support effective reuse because single object classes are too detailed and specific. Components are more abstract than object classes and can be considered to be standalone service providers. CBSE deals with the entire lifecycle of component-based software products and is a wider concept that includes component-based software development (CBSD) as a subset of lifecycle phases.

3.1 Component-Based Software Development

Although CBSD is not a novel approach, it is considered a good way to increase cost efficiency in software development and also provides increased reliability of the software when up and running, while decreasing pressure on maintenance [15]. CBSD consists of two main parts: (1) component architecture, and (2) component-based development procedure. The component architecture acts as a standard foundation for the reuse of software components [16]. If architecture is standardized than the development process is able to use components as a central aspect [16]. Component-based architecture design pattern, as shown on Figure 2, consists of several parts:

- *Component* – software implementation with one or more implemented interfaces that are imposed upon it and are implemented in a way that satisfies the component contract,
- *Component type* – small number of platform distinct types that are described with interfaces and that have specialized role in a system,
- *Component model* – made up of the component types, the interfaces of these types and the rules of the patterns of interaction which could be allowed between these component types [17],
- *Component framework* – provides a variety of runtime services to support component model.

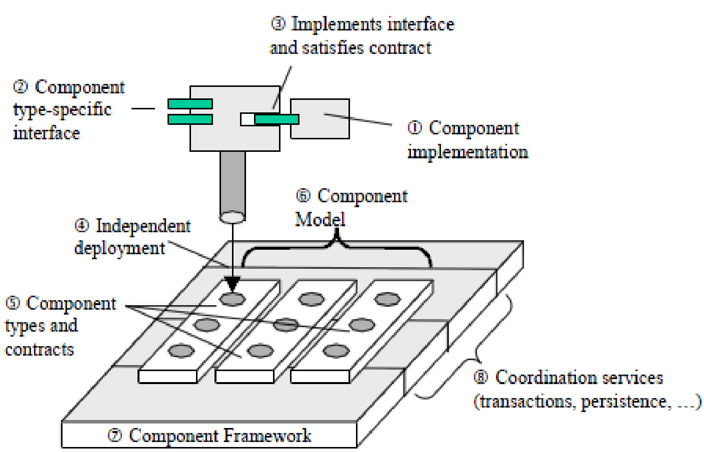


Fig. 2. The component-based design pattern [16]

Component framework acts like operating system with special purpose to provide support for components. Some examples of component models are: EJB[1] (Enterprise Java Beans), COM and COM+ from Microsoft[2] and CORBA[3] (Common Object Request Broker Architecture).

3.2 Component-Based AAC Software Development

In this paper we are proposing component-based model for developing AAC systems, and for that purpose we have designed AAC component model and specified AAC component types. Also we have designed specific component framework for our custom AAC components. Framework acts like container to all components and provides ability for components to co-exist. There are some components that are mandatory, such as e.g. “AAC symbol provider”, so we can observe them as framework functionalities because every other component must interact with some of these. Framework is responsible for interaction with device (through device operating system), while regular (non-mandatory) components are not allowed to interact with the device directly. If there is a need for using the device specific features, there should be new mandatory component, providing an interface for using that feature.

There are three different AAC component types: (1) mandatory component type, (2) visual component type, and (3) non-visual component type. Difference between these three types is in interfaces that they are implementing. Mandatory component type components are implementing functionalities that are necessary for AAC service to work. These components are:

- *Symbol component* – responsible for providing other components with the ability to fetch and work with symbols used in system,
- *HCI component* – providing other components with interfaces for user interaction through human computer interface (HCI),
- *Notification component* – used for broadcasting event notifications between components,
- *Registration component* – used for gathering data about component states. Framework must be aware of all components and their particular state.

As shown in Figure 3, mandatory components are directly interacting with the framework and they are providing interfaces for other components.

Visual type components have the ability to interact with user by presenting graphical elements on device interface while non-visual components are used for implementing non-interactive and non-visual functionalities. Interfaces that components must implement are shown on Figure 4.

---

[1] Enterprise Java Beans - http://www.oracle.com/technetwork/java/javaee/ejb/index.html
[2] Microsoft Component Object Model – http://www.microsoft.com/com/default.mspx
[3] Common Object Request Broker Architecture - http://www.corba.org/

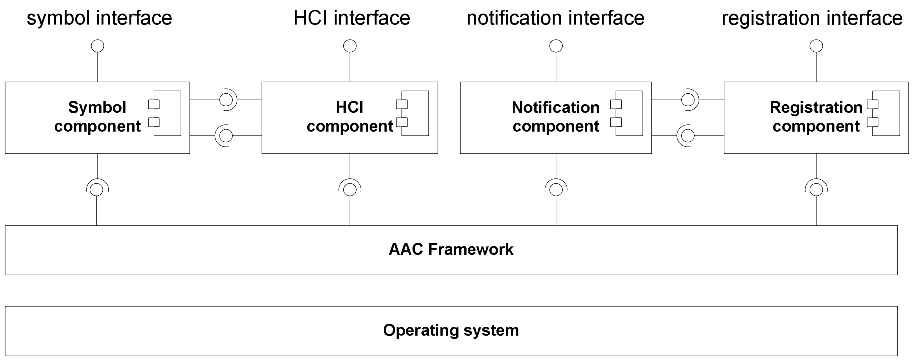


Fig. 3. AAC framework with mandatory components

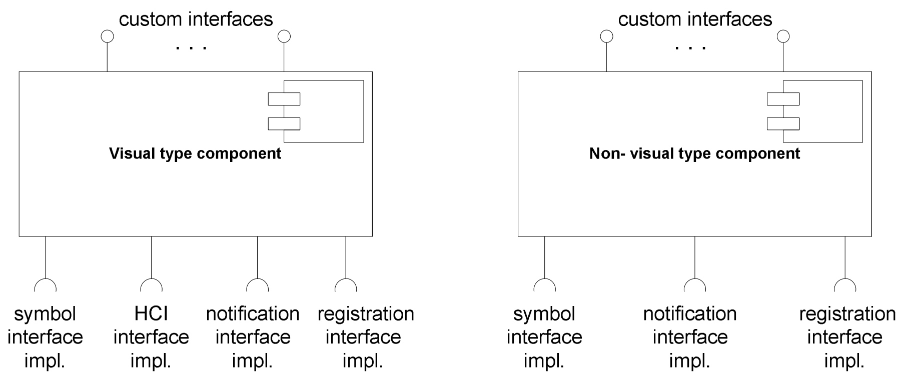


Fig. 4. Visual and non-visual component model

Beside mandatory components, which are part of the platform, there will be developed components available for building new AAC services. These components will have different common functionalities such as: selecting a symbol, fetching a symbol metadata, sending a symbol based message, symbol-to-text and text-to-symbol transformation, etc. New AAC service development process, with proposed component model is formed of 4 steps: (1) collecting requirements from user, (2) finding already developed components with the required functionalities, (3) developing new components with remaining functionalities and (4) implementing communication between components.

We claim that this process, with the reuse of developed components, provides faster development of new AAC services with no need for “reinventing the wheel”.

4 Requirements and AAC Components

Within CBSE there are two separated domains: (1) component development, and (2) building software systems from pre-existing components. It is obvious that developed components are prerequisites for successful development of component-based software and selecting appropriate components is one of early phases of component-based software lifecycle. Lifecycle can be illustrated with V-model where the first phase is requirements analysis and specification. The main goal is to analyze the possibility of implementing the solution that will meet requirements. In a component-based approach this implies that it is necessary to analyze whether these requirements can be fulfilled with available components [19].

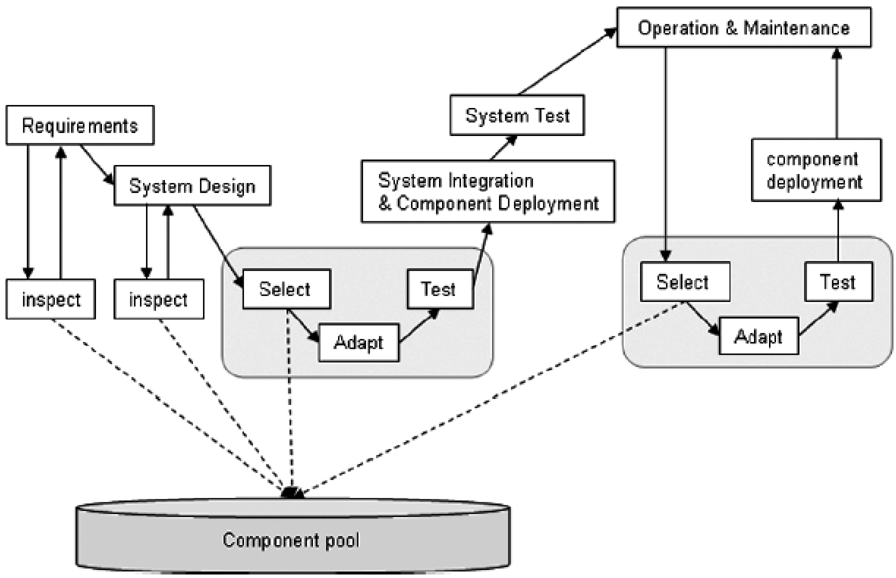


Fig. 5. A detailed V development process for CBD [17]

As stated in [19], with this approach a new process of finding and evaluating components will appear in lifecycle as shown in Figure 5. In proposed component-based model for developing AAC systems we are using ReqIF format for requirements specifications what can ease and speed-up process of finding and evaluating components. For successful finding a component with the required functionalities, developed components must be registered in component pool with metadata for describing their functionalities. We are proposing several attributes, which can be extended, for describing components. Attributes are organized as follows:

- *Input parameters* – parameters that must be submitted to component,
- *Output parameters* – parameters that are given as a result of component functionalities,
- *Functionalities* – implemented component functionalities,
- *Type* – parameter that describes whether component is visual or non-visual type.

ReqIF format contains requests specified with their attributes. While processing requests request attributes are being compared with component metadata to find components that best suits requests.

5 Conclusion and Future Work

For building a shared set of service functionality and software assets using a common means of production, software systems engineering introduced the area of software product lines in the last few decades [20]. Requirements engineering for platforms is a discipline full of challenges [21], such as incompleteness of non-functional requirements, and a large variety of users with different and complex communication needs. The first challenge can be tackled by an iterative approach while building requirements specification, with continuous inputs from the user representatives, and, in more cases, their everyday caretakers, or close relatives. The second challenge can be technically bounded by building component libraries with the components specialized for one aspect of special need (e.g. impaired hearing), but including the expertise from other scientific fields, like psychology and rehabilitation, remains crucial here. Nevertheless, some platform models for symbol based communication services are already being introduced [22] as a result of multidisciplinary collaboration [7], aiming towards the full development of platform-based AAC services.

Acknowledgement. The authors would like to thank their university professor, supervisor and a colleague Željka Car, PhD, for making useful comments regarding the proposed CBD model.

References

1. ITU-BDT: Measuring the Information Society 2011. International Telecommunication Union (ITU) – BDT, Geneva, Switzerland (2011) ISBN 92-61-13801-2
2. Heeks, R.: ICT4D 2.0: The Next Phase of Applying ICT for International Development. Computer 41(6), 26–33 (2008)
3. ISO 9241-11:1998(E): Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability. International Organization for Standardization (ISO) Standard 9241-11, 1st edn., Geneva, Switzerland (1998)
4. Cranor, L.F., Garfinkel, S.: Security and Usability – Designing Secure Systems that People Can Use. O'Reilly Media, Sebastopol (2005) ISBN 978-0-596-00827-7
5. Lloyd, L.L., Fuller, D.R., Arvidson, H.H.: Augmentative and Alternative Communication: A Handbook of Principles and Practices. Allyn and Bacon, Boston (1997)
6. Sears, A., Jacko, J.A. (eds.): The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications. Taylor & Francis Group, New York (2008)
7. Stančić, Z., Frey Škrinjar, J., Ljubešić, M., Car, Ž.: Multidisciplinary collaboration and ICT services for people with complex communication needs. In: Proceedings of the MIPRO 2011 / MEET and GVS, Rijeka, Croatia, pp. 265–270 (2011)

8. Sommerville, I.: Software Engineering, 9th edn. Addison-Wesley & Pearson Education, Harlow, England (2010)
9. Leffingwell, D.: Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise. Pearson Education, Boston (2011)
10. Belani, H., Pripužić, K., Kobaš, K.: Implementing Web-Surveys for Software Requirements Elicitation. In: Proceedings of ConTEL 2005, Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia, pp. 465–469 (2005)
11. Zielczynski, P.: Requirements Management Using IBM Rational RequisitePro. Pearson Education & IBM Press, Indianapolis, USA (2007) ISBN 978-0-321-38300-6
12. Jastram, M.: ProR, an Open Source Platform for Requirements Engineering based on RIF. In: Proceedings of the Systems Eng. Infrastructure Conference, Munich, Germany (2010)
13. Object Management Group (OMG): Requirements Interchange Format (ReqIF). v1.0.1. OMG Document Number: formal/2011-04-02 (2011)
14. Abrial, J.-R.: Modeling in Event-B: System and Software Engineering. Cambridge University Press, UK (2010) ISBN: 9780521895569
15. Szyperski, C.: Component Software: Beyond Object-Oriented Programming, 2nd edn. Addison-Wesley, Boston (2002)
16. Ning, J.Q.: A Component-Based Software Development Model. In: Proceedings of the Annual Int. Computer Software and Applications Conference, pp. 389–394 (1996)
17. Gao, J., Tsao, H., Jacob, S., Wu, Y.: Testing and Quality Assurance for Component-based Software. Artech House, Boston (2003)
18. Bachmann, F., Bass, L., Buhman, C., Comella-Dorda, S., Long, F., Robert, J., Seacord, R., Wallnau, K.: Volume II: Techincal Concepts of Component-Based Software Engineering. Techincal Report CMU/SEI-2000-TR-008, Carnegie Melon University, USA (2000)
19. Crnković, I., Chaudron, M., Larsson, S.: Component-based Development Process and Component Lifecycle. In: Proceedings of the ICSEA 2006. IEEE, French Polynesia (2006)
20. Gomaa, H.: Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures. Addison-Wesley, New York (2004)
21. Berenbach, B., Paulish, D., Kazmeier, J., Rudorfer, A.: Software & Systems Requirements Engineering: In Practice. McGraw-Hill Osborne Media, New York (2009)
22. Car, Ž., Vuković, M., Vučak, I., Pibernik, J., Dolić, J.: A Platform Model for Symbol Based Communication Services. In: Proceedings of the 11th International ConTEL 2011, pp. 141–147. Graz University of Technology, Graz (2011)

e-Accessible Service System: Calibrator and Communicator

Ivan Blagajic, Ivan Semanjski, Tomislav Saric, Zeljka Janda-Hegedis, Marin Vuković, and Željka Car

University of Zagreb, Faculty of Electrical Engineering and Computing, Croatia
{ivan.blagajic,ivan.semanjski,tomislav.saric, zeljka.janda-hegedis,marin.vukovic,zeljka.car}@fer.hr

Abstract. Augmentative and Alternative Communication (AAC) includes various communication methods used by people with speech and language impairments, and is often aided by electronic devices such as smartphones or tablets. Since skills and limitations vary for each user, it is possible to create an interface on these electronic devices that would adjust to user's needs in ways that would significantly improve usability and ease of use. Paper describes e-Accessible service system containing Calibrator and Communicator services. Calibrator measures user's preferences and limitations regarding user interface through calibration process and calculates UI parameters adapted to the particular user. The parameters are forwarded to the Communicator service, whose main purpose is to supplement speech or writing for people with complex communication needs.

Keywords: alternative and augmentative communication, symbols, automatic interface adaptation, calibrator, communicator.

1 Introduction

Augmentative and Alternative Communication (AAC) includes various methods of communication used by people with complex communication needs. Depending on whether AAC uses any external tools, it can be divided into two categories: unaided and aided. While unaided AAC relies solely on messages transmitted by user's body, aided AAC uses additional electric or non-electric devices to facilitate communication for people with speech and language impairments. Low-tech aids are usually very simple communication boards, while high-tech ones (e.g. dedicated devices designed solely for AAC or non-dedicated devices like personal computers, tablets, smartphones, etc.) offer many opportunities to ease application of AAC. They are able to send/ retrieve electronic messages, store large amount of information and present it in a way that will help users communicate faster and more efficiently [1].

AAC methods are based on the use of graphic and textual symbols (rather than written words alone) to represent certain objects, actions, or concepts. Besides the communication, symbols are beneficial for learning and development of complex expressions [2].

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 241–250, 2012.

AAC Communicators are devices that allow people with complex communication disabilities to attract attention, make requests, choose or reject something. Each symbol on device has its own audio record that is played when symbol is pressed. They vary from simple message communicators, enabling users to make a choice between two or more predefined answers, and more complex devices used for selecting symbols, letters or words in order to generate meaningful phrases. Typical communicator (example shown in Figure 1) has limited capabilities because it is based on inserting cards with symbols into provided slots thus limiting the number of symbols shown. Depending on device, duration of all audio records can be from few seconds to few hours.

Fig. 1. Example of conventional AAC Communicator [4]

Each AAC user has its own unique set of abilities and needs and these sets significantly differ. In order to customize service user interface (UI) to the particular user it is possible to apply several approaches: (i) manually changing default UI settings by advanced user (parent, teacher or personal assistant). This can be exhaustive process with uncertain outcome and, due to limited user communication possibilities, one cannot be sure that the adaptation is really successful; (ii) creating several categories of users and determining initial interface settings for each category separately. Since one user can have multiple limitations it is very difficult to create such categories and additional manual adaptation is required to conform to particular user's abilities good enough, which turns back to problem stated in (i).

As a solution to the problem of initial UI adaptation, e-Accessible service system is designed and implemented. The system and its services are described in further text.

2 Description of Calibration and Communication System

E-Accessible calibration and communication service system comprises two services: Calibrator and Communicator. Calibrator implements functionality of initial UI adaptation in the form of interactive game and consists of four levels that measure different user preferences. Measured data is used to calculate parameters assigned to user's online profile and is forwarded to any other e-Accessible symbol based service, such as Communicator. Communicator service enables users to communicate by selecting particular symbols. Both services are implemented for touchscreen tablet computers, Calibrator for Android and Communicator for iOS.

2.1 System Architecture

System architecture consists of three distinct entities: two different services and server with database. The services communicate with the server and are developed on different

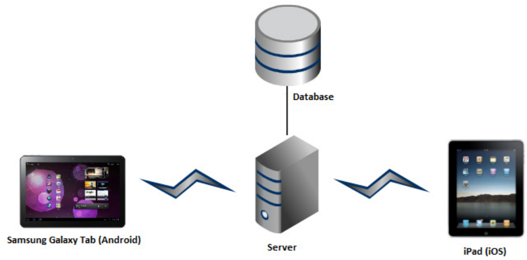


Fig. 2. e-Accessible service system deployment

operating systems to emphasize the portability of e-Accessible service system and of the overall calibration approach. Calibration process is achieved through several tests on Android based tablet device. Service system deployment is presented in the Figure 2.

Usage of e-Accessible service system is designed as follows. At first use, the user runs calibration service on Android tablet and goes through the calibration process. After its completion, all measured data is sent to server and processed. Processing data results in interface parameters which are stored in relational database. Stored parameters are then used to organize Communicator service interface.

2.2 Use Cases

Interactions between actors and services for Calibrator and Communicator are presented in Figure 3.

As shown in Figure 3a, calibration service is accessed both by user and advanced user (e.g. parents, teachers, caregivers and other professionals). Role of advanced user is entering and editing personal information as well as setting test parameters for particular user.

Figure 3b shows how actors (user and advanced user) interact with Communicator service. Advanced user registers or logs in as symbol user and changes interface parameters manually through a settings menu, if necessary.

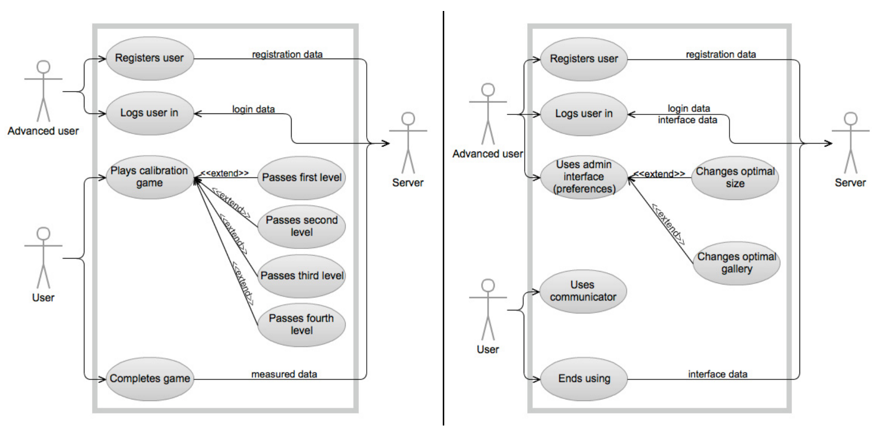


Fig. 3. Calibration service and Communicator use case diagram

3 AAC Calibrator

Calibrator is an e-Accessible service that consists of several short tests whose task is to determine which parameters are most convenient for particular user. Parameters are symbol size, position on the interface, gallery of symbols selected and background-symbol contrast. There are four different tests designed as short games. In every game different symbols are displayed in sequence on interface and user's task is to touch them. Symbols have different sizes, backgrounds and positions on the interface, so by measuring time and precision of a user touch the service can calculate parameters mentioned above. These four tests are described in the following subsections.

3.1 Determining Optimal Size of Symbols

First test is about determining optimal symbol size. In this context optimal size means that user can see symbol well and clear and that he/she can select it successfully and precisely. Measurements are conducted by sequentially displaying symbols of different sizes. This test measures time required for user to touch the symbol. Default size of the symbol is set to a size of a square with one side equal to one third of wider part of screen, while other sizes are obtained by decreasing its preceding size by 15%. Advanced user can adjust following settings for this test:

- Number of levels (number of different sizes from 1 to 6)
- Number of symbols per level
- Test timeout

Besides time, this test is constrained by user's ability to precisely select each provided symbol. If user misses more than half of displayed symbols on a certain "difficulty" level, test is finished and application proceeds to the next test.

Data collected in this test is related to the number of successfully completed levels.

3.2 Determining Appropriate Positions on the Interface

Second test measures suitable positions of elements on the interface. The aim of this test is to identify areas of the interface where user touches the symbol without any problems. Measurements are conducted by sequentially displaying symbols on predefined positions on the interface (as shown in Figure 4). Every symbol displayed in each position needs to be touched twice to be marked as hit (symbol appears in two different shapes at the same position to motivate the user to select it twice). Each state is limited with the timeout period. Successful outcome of this test is touching the symbol twice in defined time interval, while unsuccessful outcome is reaching the predefined maximum number of misses per symbol state or timeout for that state.

Result of this test is set of successfully touched positions and average time taken by user to touch the symbol positioned on specific coordinates. Advanced user can adjust following settings for this test:

- Maximum number of misses per symbol state
- Timeout of symbol displayed

Fig. 4. Game for determining appropriate positions

These results can be used to determine suitable positions for toolbar, menu bar, symbol gallery or other elements on UI of any other service.

3.3 Determining Most Appropriate Symbols Gallery

The aim of third test is to determine a gallery whose symbols are most suitable for the user. Measurements are conducted by displaying required term and group of symbols with each symbol belonging to a different gallery, but depicting the same term (as shown in Figure 5).

This test records which symbols from which galleries are chosen by the user. Audio record of the required term can be played by clicking a special icon on the interface.

In this test there is also a time limit defined, in which symbol has to be touched. Therefore, advanced user can adjust following settings for this test:

- Number of different terms
- Timeout of symbol display

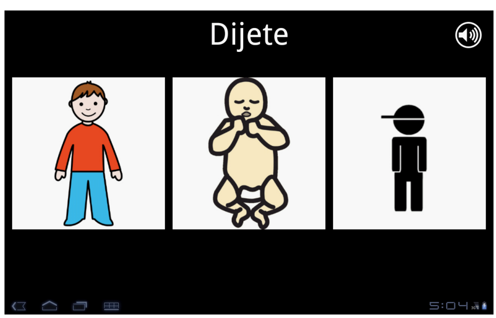


Fig. 5. Game for determining most appropriate gallery of symbols

Result of this test is name of gallery from which user has picked the most symbols and therefore showed that this gallery has, in average, most understandable set of symbols.

3.4 Determining Optimal Contrast

Last test determines optimal contrast between symbol and background. Symbols are represented as framed letters and numbers. Aim of this test is to determine the most appropriate contrast between the color of UI background and symbol colors. Measurements of one combination of background and symbol color are conducted by displaying defined number of symbols on different screen positions in several iterations. Test also measures time needed for user to touch symbol on a different background color.

Advanced user can adjust following settings for this test:

- Number of levels per color combination
- Number of symbols per level
- Timeout for displaying group of symbols

Result of this test is combination of background and symbol color that is the most appropriate for the user.

4 AAC Communicator

There is a number of different communicators in the market today, both stand-alone devices as well as tablet and smartphone services, but none of them support automatic adjustment of user interface. However, few notable solutions are listed below:

- Lighthawk - A symbol-scanning communicator that contains three levels, each with eight sections for displaying symbols. First and second level can hold a soundtrack 7.5 seconds long, while third one can store records 15 seconds long. [6]
- SpringBoard Lite - It can display 4, 8, 16 or 32 symbols on the screen, has simplified and improved tools, both male and female audio track available, is portable and it has enough memory to store a total of 500 minutes of audio recording. [7]
- Superhawk Plus - Touch screen device that displays 72 icons on the screen. User is able to form complex sentences. The longest sound recording can last for 72 minutes. [8]
- JA-KOM - This web communicator provides the opportunity to learn basic mathematical operations and connect to some popular sites: Google, Facebook and YouTube. [9]
- TapToTalk - Android and iOS based communicator that can store files and save custom sounds. Allows organization of images into albums (more than 12 albums, each a few hundred files). [10]

Communicator developed as part of e-Accessible service system runs on iOS tablet devices (iPad and iPad 2) and can show 6, 12 or 20 symbols at once, using one of three pre-defined open source galleries (Arasaac, Mulberry and Sclera).

Communicator is designed to support multiple users on one device as well as one user on multiple devices. There are three different ways to start using Communicator.

Users that have passed through calibration process and already have their usernames can start using Communicator immediately. Other users should either create their profile through registration process or use Communicator in offline mode, with manual UI settings.

4.1 Adjusting User Interface and Preferences

After the user signs in, Communicator retrieves calibration data from the server and adjusts its UI using symbol size and gallery that are determined by the calibration process.

Given the fact that there are only three different symbol sizes defined on Communicator, interface parameter that describes size number is applied in the following way:

- If interface parameter is 1, 6 large symbols are shown on single page
- If interface parameter is 2, 12 medium symbols are shown on single page
- If interface parameter is 3 or larger, 20 small symbols are shown on single page

This represents how different services can use interface parameters in different ways, depending on their purpose. After experimenting with various symbol sizes it was concluded that UI should display 6 to 20 symbols on the screen.

Regarding symbol galleries, there are three different galleries available to choose from: Arasaac, Mulberry and Sclera. When Communicator loads its main interface, all symbols shown belong to the most appropriate gallery. So, for the same word or expression there are at least three different symbols, but only one is shown and it is from gallery that user understands best.

Using most appropriate position and optimal contrast had no application in Communicator, but it can be very helpful in other services for users with limited motor skills or those that are colorblind. While application of contrast is pretty straightforward (similar to most appropriate gallery of symbols), multiple positions can be suitable for user so one of them should be chosen by some criteria (e.g. does not cover or interfere with other interface elements, etc.).

Advanced user can access manual settings screen through '*Slide to unlock*' screen slider to avoid accidental opening of this screen. Settings contain editable user personal information and interface options. If any interface parameter is manually changed in the settings, UI is immediately adjusted, and online user profile is updated with new parameters. Old data is not deleted, so historical data can be accessed any time and user's progress can be monitored.

5 Case Study

Case study of e-Accessible service system usage involves two users (user A and user B). User A goes through calibration process first and fails to successfully select symbols with size 2, thus resulting with largest symbol size chosen as most appropriate. He also shows best understanding of Mulberry gallery of symbols by choosing Mulberry symbols most number of times. When user A is signed in to Communicator, interface is adjusted as shown in Figure 6.

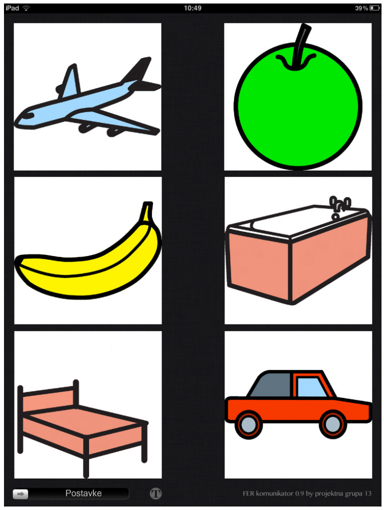


Fig. 6. User A's interface after calibration

User B's profile is created on Communicator where medium symbol size and Arasaac gallery are manually set as defaults by advanced user. After user's registration, he/she is presented with Communicator UI as shown in Figure 7.

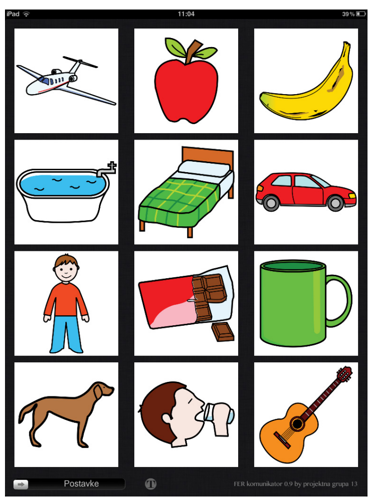


Fig. 7. User B's interface after registration

User B has no problem with selecting displayed symbols but shows some difficulties understanding them. Therefore, he also goes through calibration process. After signing-in to Calibrator with previously created profile on the Communicator, user B passes first test with 100% accuracy and in the third test selects Sclera symbols the most number of times. After the calibration process is finished, user B sings-in to Communicator again, but the interface now looks as shown in Figure 8, corresponding to his/her preferences and possibilities.

Fig. 8. User B interface after calibration

6 Conclusion

Considering fast growing smartphone and tablet markets, it is very likely that these devices will become very affordable in near future. The proposed e-Accessible service system consisting of Calibrator and AAC Communicator services facilitates application of AAC in everyday communication. It is important to emphasize that the implemented calibration process is generic and can be applied for any other service, beside Communicator. Future work will focus on improvements and upgrades listed below:

- interface for sending e-mail/SMS messages
- instant messaging service
- symbols to text (and vice versa) processor
- using services with symbols as a remote controller for various purposes
- specialized plugin for major Internet browsers
- general interface adaptation of any touch screen device (from smartphones to big TV screens)
- adaptation of popular children games' interfaces

Such services would significantly improve life quality of persons with complex communication needs and provide their better inclusion in the society.

Acknowledgments. This work was carried out within development project "ICT systems for Persons with Complex Communication Needs", supported by University of Zagreb Development Fund.

References

1. ISAAC – The International Society for Augmentative and Alternative Communication, http://www.isaac-online.org/
2. Car, Ž., Vuković, M., Vučak, I., Pibernik, J., Dolić, J.: A Platform Model for Symbol Based Communication Services. In: Proceedings of the 11th International Conference on Telecommunications ConTEL 2011. Graz University of Technology, Graz (2011)
3. Stančić, Z., Frey Škrinjar, J., Ljubešić, M., Car, Ž.: Multidisciplinary collaboration and ICT services for people with complex communication needs. In: MIPRO 2011 Proceedings of the Conferences Microelectronics, Electronics and Electronic Technology, Rijeka (2011)
4. Talara 32, Spectronics website, http://www.spectronicsinoz.com/
5. e-Inclusion, Europe's Information Society Thematic Portal website, http://ec.europa.eu/information_society/activities/einclusion/
6. Lighthawk, Adamlab website, http://www.adamlab.com/
7. SpringBoard Lite, Prentrom website, http://www.prentrom.com/springboardlite/
8. Superhawk Plus, Adamlab website, http://www.adamlab.com/
9. JA-KOM Portal website, http://www.aachr.com/ja-kom/
10. TapToTalk website, http://www.taptotalk.com/

Evaluation of Mainstream Tablet Devices for Symbol Based AAC Communication

Jurica Dolic, Jesenka Pibernik, and Josip Bota

University of Zagreb, Faculty of Graphic Arts, Croatia
{jurica.dolic,jesenka.pibernik,josip.bota}@grf.hr

Abstract. Rapid evolution of mobile devices enables the today's user to access the content and technology previously reserved for multiple devices in one compact, portable package. The introduction and popularization of smartphones, tablets and e-readers changed the way that the users communicate, consume and create content. This paper examines the potential of using such devices for symbol-based Augmentative and Alternative Communication. We have proposed a model of an adaptive symbol-based AAC application for mainstream mobile devices. In respect of the requirements of the model we have examined technical possibilities and limitations of specialized and mainstream tablet devices available. Via a preliminary research of potential symbol-based AAC users in Croatia we have investigated the current state, needs and requirements regarding the use of digital AAC applications.

Keywords: Augmentative and Alternative Communication, Mobile devices, Tablets, Symbols.

1 Introduction

Mobile devices have become an important part of our daily lives. Having started as devices that created freedom in voice communication, they have developed into powerful multimedia platforms. According to the latest reports, smartphone penetration in the U.S. market was 44% of all mobile phone devices in 2011 [1].

In times when a large part of social interaction is happening online, whether via e-mail, online forum and chat systems or social network websites, users with special needs are often excluded because of technological and functional limits imposed by the dedicated solutions they use. This digital exclusion is preventing individuals with disabilities from having socially active and independent lifestyle [2]. AAC users think that the technology and AAC solutions "must support full participation in all aspects of 21st century life" [3]. In the year 2005 the European Union started a "i2010" initiative, which promotes "e-Accessibility" with the goal of enabling access to ICT applications, services and devices to people with disabilities [4].

The most important part of the HCI process using symbol based AAC programs and services on touch screen mobile devices is happening on the very screen – from symbol display and browsing, to the selection process and word construction. It is therefore essential that the design and display of symbols and user interface lend themselves to screen specifications. A number of portable devices with touch screen interface and mobile operating system – from mobile phones and tablets, to e-readers

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 251–260, 2012.

and digital cameras, many of them using different display technologies and sizes, is growing constantly. Even in the same category of devices, the manufacturers are using different screen sizes and display technologies to diversify their product lines or to cut costs. Unlike the dedicated AAC solutions, where the manufacturer has full control over device specifications and software implementation, AAC programs and services for mainstream mobile devices should demonstrate functionality on a wide range of display types, sizes, resolutions, colour spaces and refresh rates. Due to such a variety, there is a need for a different approach to design and the development of symbol-based AAC programs and services for these devices.

2 Model of Adaptable Symbol Based AAC Application for Mainstream Mobile Devices

The proposed model is an adaptive system that will adjust the display of symbols and user interface to the specifications of the device, but also to the capabilities and preferences of the users (Fig. 1).

The system will use specially developed symbols which will enable the change in size without significant loss in quality. Each symbol will have three different visual variations – one using full 16 million colour palette, the other using limited colour palette and a black-and-white variation. Each variation will be able to feature animation of graphic objects. The user would be able to use symbols from an online repository or store them for local use.

Upon the first use of the system, the user or his assistant would set their preferences regarding symbol type, size, GUI layout and the use of colours and animation. The system would then upload the preferences to an online database.

After the installation on a new device, the program will automatically set minimum symbol size taking into consideration screen resolution, maximum number of displayed symbols based on the screen size and minimum symbol size, colour reproduction based on the screen colour space specifications and the possibility of animation regarding screen refresh rate and the processing power of the device. These settings will then be combined with user preferences to create the final display of the symbol-based AAC application on the device. User preferences may not exceed the boundaries that the program determined in respect of the specifications of the system.

When the application is used with another device, it will automatically optimize its display to suit the screen specifications and user preferences. This will enable the users with multiple mobile devices to use same AAC solution on every device without having to set it up manually every time they use it. Furthermore, the program will work on most upcoming devices with the same mobile OS, thus enabling AAC users to follow technological advancements easily.

Since such system largely depends on user preferences, requirements, possibilities and expectations regarding the use of such system with mainstream mobile devices, it is important to include the user in the design process using principles of User Centred Design (UCD). The importance of user feedback is often overlooked when designing such services, despite their requests to participate in such projects [3].

For this system to work, it is important that the specifications of mobile devices used meet some minimal requirements. Since they should offer a similar level of functionality compared to dedicated mobile devices, they should have a similar range of screen sizes, and similar or better hardware specifications. Also, since this is an adaptable system, it is important that the mobile operating system used works on several different devices and that it allows access to different hardware components. All this requires a more detailed insight into technical capabilities of dedicated and mainstream mobile devices.

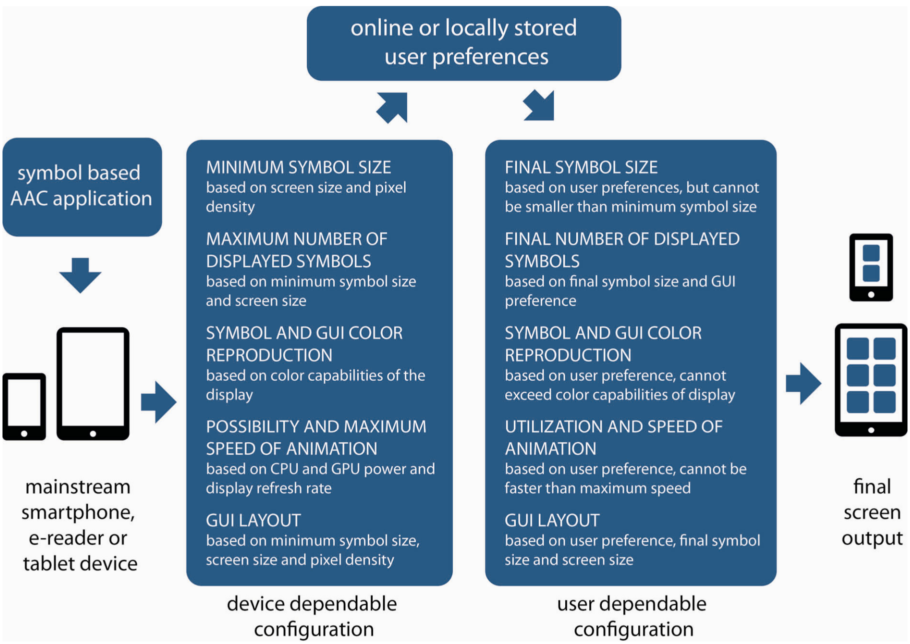


Fig. 1. Model of adaptive symbol based AAC application for mainstream mobile devices

3 Analysis of Dedicated AAC and Mainstream Mobile Devices

Users of symbol-based AAC systems can use dedicated and non-dedicated electronic devices to help them facilitate the communication [5]. While non-dedicated devices are mainstream technologies that use AAC software, dedicated devices are built specifically for this kind of user base.

In order to get a better insight into the capabilities of current dedicated AAC mobile devices, we have made an evaluation of their technical characteristics and compared them to mainstream tablet devices of similar screen sizes (Table 1, 2).

The evaluation of the device has been made for the U.S. market because it offers the widest range of dedicated AAC devices. The majority of evaluated mainstream devices are available worldwide, while dedicated AAC devices have limited availability and are distributed mainly on larger North American and European markets.

Table 1. Evaluation of dedicated AAC and mainstream tablet devices with screen sizes form 7 to 9 inches

	mainstream tablet devices			dedicated AAC tablet devices						
Device name	**Samsung Galaxy Tab 8.9**	**Barnes&Noble Nook Tablet**	**Samsung Galaxy Tab 7 Plus**	**Springboard Lite**	**Vantage Lite**	**Jabbla Mobi 2**	**Tobii C8 Communication aid**	**Jabbla Zingui**	**Words+ Sam Tablet SM1**	**Dynavox V+**
Screen type	LCD	IPS LCD	LCD	LCD	LCD	LCD	LCD	LCD	LCD	LCD
Screen size	8.9 in	7 in	7 in	7 in	8.7 in	8.4 in	8.4 in	8.4 in	8.4	8.4
Resolution	1280 x 800 px	1024 x 600 px	1024 x 600 px	800 x 480 px	800 x 600 px	800 x 600 px	800 x 600 px	640 x 480 px	800 x 600 px	800 x 600
Pixel density	170 ppi	170 ppi	170 ppi	133 ppi	115 ppi	119 ppi	119 ppi	91 ppi	119 ppi	119 ppi
CPU type and clock speed	dual-core, 1 GHz	T.I. OMAP 4 dual-core, 1 GHz	Exynos dual-core, 1.2 GHz	N/A	N/A	AMD Geode LX800, 500MHz	Intel Core Duo U2500	Marvell Xscale PXA	AMD Geode, 800MHz	Intel Atom, 1.6 GHz
RAM size	1 GB	1 GB	1 GB	N/A	N/A	1 GB	2 GB	128 MB	128 MB	1 GB
Memory size	16 GB (32 GB optional)	16 GB	32 GB	N/A	N/A	8 GB	60 GB	1 GB	2 GB	80 GB
Camera	Yes, 2	No	Yes, 2	No	No	No	Yes	No	No	No
Camera resolution	rear – 3 MP front – 2 MP	/	rear – 3 MP front – 2 MP	/	/	/	0.3 MP	/	/	/
Connectivity	Wi-Fi Bluetooth USB 2.0 microSD card head phone (3G/4G optional)	Wi-Fi USB 2.0 microSD card headphone	Wi-Fi Bluetooth IR control USB 2.0 microSD card head phone (3G/4G optional)	IR control USB SD card microphone head phone switch (Bluetooth optional)	Bluetooth IR control USB SD card speakers head phone switch	Wi-Fi IR control USB switch (Bluetooth optional)	USB SD card LAN switch port head phone (Wi-Fi, Bluetooth optional)	USB switch (IR control optional)	USB switch	Wi-Fi Bluetooth IR control USB switch head phone microphone
GPS	Yes	No	Yes	No	No	No	No	No	No	No
Integrated Speakers	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Integrated Microphone	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes
Battery life	9 h	9 – 11.5 h	8 h	6 - 8 h	5 – 7 h	N/A	6 h	N/A	4 – 6 h	9.5 h
Operating system	Android OS, v3.1	Android OS	Android OS, v3.2	Unity Language System	Unity Language System	Windows XP home	Windows 7	Windows CE	Windows CE	Windows 7
Device dimensions (WxHxD)	231 x 157 x 8 mm	205 x 127 x 12 mm	194 x 122 x 10 mm	185 x 185 x 45 mm	220 x 234 x 41 mm	327 x 245 x 40 mm	239 x 201 x 38 mm	215 x 165 x 38 mm	267 x 220 x 51 mm	230 x 200 x 70 mm
Device weight	448 g	400 g	343 g	1133 g	1530 g	1750 g	1800 g	1080 g	1769 g	2040 g
Price	$ 400	$ 250	$ 449	$ 2595	$ 7495	$ 7795	$ 6395	$ 4095	$ 7095	$7800

3.1 Dedicated Mobile Devices for Symbol Based AAC Use

With rapid advancement of mobile technology, dedicated AAC devices keep failing to follow the trends and technologies found in mainstream mobile devices. One of the main problems is that dedicated mobile AAC devices are not providing access to multiple communication functions and electronic tools, as opposed to mainstream tablets and smartphones [6]. They often lack usable hardware components like a camera to enable the users to create their own graphic signs by utilising photographs taken or GPS module, so they could find their way around if they are lost or send information about their current location. Also, since they often use customized operating systems and proprietary software, there are not any options for expanding the capability of the device by installing additional applications made by development community. This lack of convergence hinders user's ability of accessing and consuming media and can limit his communication and socialization ability.

The main interactive part of these devices, the touch sensitive display, is also not at the same level as the current mainstream offerings, often having limited colour reproduction and lower resolutions. Snellan 20/20 acuity for viewing distance of 50,8 cm requires screen pixel density of 172 ppi (pixels per inch) [7] but it is normal for children and young adults to have 20/16 acuity and they should therefore be able to resolve details on screens with higher pixel density [8]. Some authors claim that, by varying grey levels of pixels making up the characters, pixel density as low as 140 ppi can be used to display text with performance equivalent to that of a printed page [9, 10]. Unfortunately, from the evaluation of dedicated AAC mobile devices (Table 1, 2) it can be seen that all devices use lower pixel densities, some even going as low as 91 ppi. These low pixel densities can limit user performance and cause discomfort [11].

Table 2. Evaluation of dedicated AAC and mainstream tablet devices with screen sizes form 9 to 12 inches

	mainstream tablet devices			dedicated AAC tablet devices		
Device name	**Apple iPad Wi-Fi +3G**	**Samsung Galaxy Tab 10.1 4G**	**ASUS Transformer Prime**	**Words+ Conversa**	**DynaVox Maestro**	**Tobii C12 AAC device**
Screen type	IPS LCD	LCD	Super IPS LCD	LCD	LCD	LCD
Screen size	9.7 in	10.1 in	10.1 in	12 in	10.4 in	12 in
Resolution	1024 x 768	1280 x 800	1280 x 800	N/A	1024 x 768	1024 x 768
Color range	132 ppi	149 ppi	149 ppi	N/A	123 ppi	107 ppi
CPU type and clock speed	Apple A5 dual-core, 1 GHz	Cortex A9 dual-core, 1 GHz	Cortex A9 quad-core, 1.3 GHz	Intel Core 2 Duo, 1.5 GHz	Intel Atom, 1.6 GHz	Intel Core Duo U2500
RAM size	512 MB	1 GB	1 GB	2 GB (4 GB optional)	1 GB	2 GB
Memory size	32 GB (64 GB optional)	32 GB (64 GB optional)	32 GB (64 GB optional)	120 GB	64 GB	60 GB
Camera	Yes, front and rear	Yes, front and rear	Yes, front and rear	No	No	Yes
Camera resolution	rear - 0.7 MP front - 0.3 MP	rear - 3 MP front - 2 MP	rear – 8 MP front – 1.2 MP	/	/	0.3 MP
Connectivity	Wi-Fi Bluetooth 3G head phone dock connector	Wi-Fi Bluetooth USB microSD card 3G/4G head phone port	Wi-Fi Bluetooth microSD card head phone microphone dock connector	Wi-Fi LAN Modem USB IEEE 1394a	Wi-Fi Bluetooth IR control USB head phone microphone switch	LAN USB SD card switch head phone (Wi-Fi, Bluetooth optional)
GPS	Yes	Yes	Yes	No	No	No
Integrated Speakers	Yes	Yes	Yes	Yes	Yes	Yes
Integrated Microphone	Yes	Yes	Yes	Yes	Yes	Yes
Battery life	9 – 10 h	9 h	12 h	3.5 h (7h with extended battery)	3 h (9.5h with extended battery)	5 h
Operating system	iOS	Android OS	Android OD, v3.2	Windows XP Home	Windows 7	Windows 7
Device dimensions (WxHxD)	241 x 186 x 9 mm	257 x 175 x 9 mm	263 x 181 x 8 mm	293 x 220 x 89 mm	270 x 216 x 47 mm	312 x 264 x 46 mm
Device weight	613 g	567 g	586 g	2041 g	1250 g	2900 g
Price	$ 729	$ 630	$ 500	$ 8295	$ 8025	$ 7395

Since manufacturers base their software implementation on the PC operating systems, which are less CPU and memory efficient and are not fully optimized for touch input, these devices are larger and significantly heavier than comparable mainstream devices, often having shorter autonomy time (Table 1, 2).

Despite a somewhat older technology, mobile AAC devices can be more than 15 times more expensive than mainstream tablet devices (Table 1, 2), which makes them less accessible to users in countries where they are not subsidized by medical institutions. Furthermore, the lack of support for multiple languages makes these devices unattractive to users outside the English speaking countries.

3.2 Mainstream Mobile Devices

While early mainstream mobile devices lacked touch screen input and had low resolution screens and limited colour reproduction, today they are sophisticated devices that, among other capabilities, are able to display detailed graphics on high resolution displays. With the advancement of technology and the introduction of mobile platforms utilizing dual-core and quad-core Central Processing Units (CPU) combined with powerful Graphical Processing Unit (GPU), the today's mainstream mobile devices are even capable of running complex 3D applications, filming and reproducing high definition video and doing tasks previously reserved for personal computers.

Whereas smartphones usually have smaller screens than the majority of symbol-based AAC devices, mainstream tablets have similar screen sizes as dedicated AAC devices (Table 1, 2), and therefore more potential to be used as a non-dedicated AAC device.

Unlike dedicated AAC devices, which use mainly customized PC operating systems, mainstream smartphone and tablet devices use operating systems developed and optimized for mobile devices (mobile OS).

By October 2011, Android and iOS mobile operating systems had the biggest U.S. smartphone market share [12], with Android devices being dominant on the worldwide market with 52.5% market share [13]. Unlike current PC operating systems, newer mobile operating systems like iOS and Android are optimized for touch screen interface and run smoothly on lower powered processors, with low quantities of Random Access Memory (RAM). Both, iOS and Android, as well as several other mobile operating systems, offer support for software programs and services called “applications”. Applications can be used to add extended ACC functionality to the device, essentially turning them into a non-dedicated AAC device [14]. There are several applications available for symbol based AAC users, but only few complete solutions for communication aid, intended primarily for English speaking markets [15]. Some mobile operating systems have several integrated accessibility options, but they do not cover users which primarily use symbols in communication [16].

Most of new tablets and smartphones have an integrated touch screen, speaker, microphone, digital camera, Bluetooth, Wi-Fi, 3G and GPS modules (Table 1, 2). The majority of mobile operating systems allow access of the application to all of these modules, so AAC applications could enable the user to access all technological abilities of the device. A potential problem might be that several mobile OS developers limit access of the application to change the home screen or to change visual properties of operating systems Graphical User Interface (GUI). This could be a problem for some users, because it requires them to enter menus and start the application using standard GUI. On some mobile operating systems this can be bypassed by using customizable graphic shortcuts on starting screen, eg. “widgets” on Android. Since Android is an open-source OS, device manufacturers have the ability to modify the visual and functional aspects of the GUI significantly, giving them the potential to develop a dedicated AAC device running a mobile OS with the ability of expanding the functionality through additional applications.

Mainstream mobile devices lack switch input for providing alternative input methods using specialized AAC devices, but since applications have access to the Bluetooth module, AAC application developers have the ability to add support for

Bluetooth-enabled switch input devices or other specialized input devices like various keyboards or mice, providing even better accessibility and increasing interoperability between AAC devices and mainstream technologies [6]

4 User Survey

In order to get a better insight into problems that Croatian AAC users are facing when using the solutions available on the market and so as to try to understand better their wishes, needs and requirements, an online questionnaire has been distributed to the parents of current and potential symbol based AAC system users in the period from 15th April to May 27th 2011. The questionnaire consisted of several general questions about children, followed by questions concerning technologies and programs that children use in their daily activities, and regarding the technologies and programs the parents would like their children to use. There were a total of 15 respondents, 11 having a male and 4 having a female child with a disability aged between 5 and 21. Only 3 out of 13 respondents answered that their child uses a dedicated AAC device, only one of whom using a communicator. Parents identified the small size of letters (7 answers) and symbols (6 answers) as main problems with AAC programs that their child is (or was) using (Fig. 2). Foreign language was mentioned third with 5 answers. Only two parents consider that their children have not encountered any problems.

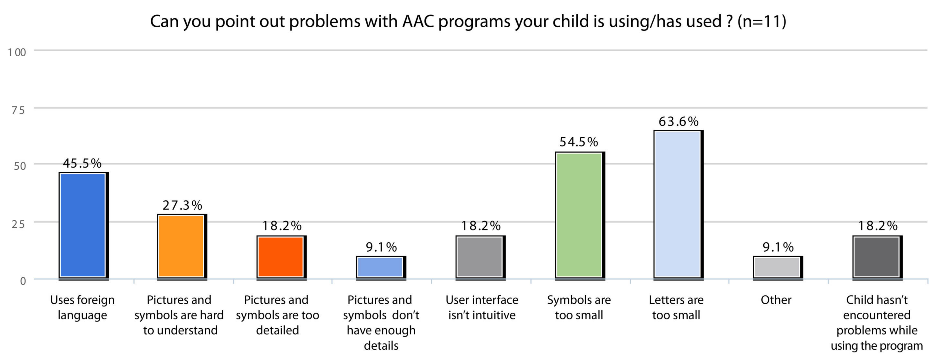


Fig. 2. Problems that users are having with current AAC programs

Currently, children mostly use AAC programs and web pages for entertainment (8 answers) and education (6 answers) and none of the children uses it for online communication (Fig. 3). All respondents want their children to use AAC programs and web pages in the future, demonstrating increased interest for the use in education (10 answers), entertainment and book-reading, but also for online communication, as a communication aid, for drawing and as a navigation aid (Fig. 3).

All parents own a PC with internet access, with only one third of them having at least one smartphone device in their household (Fig. 4). None of the respondents owned a mainstream tablet device.

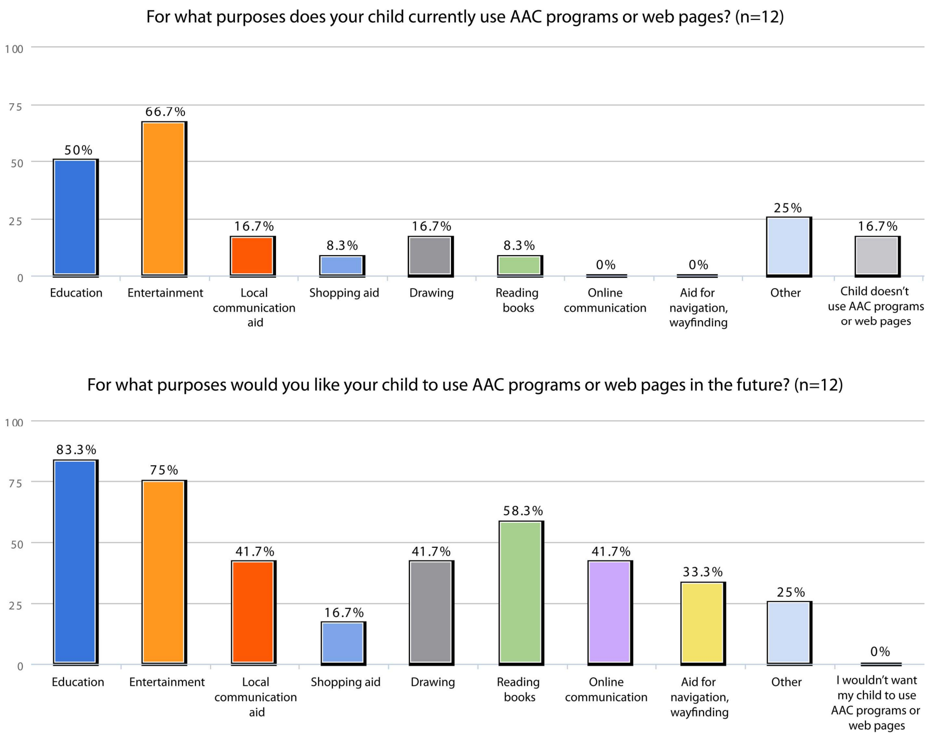


Fig. 3. Current and desired use of AAC programs and web pages

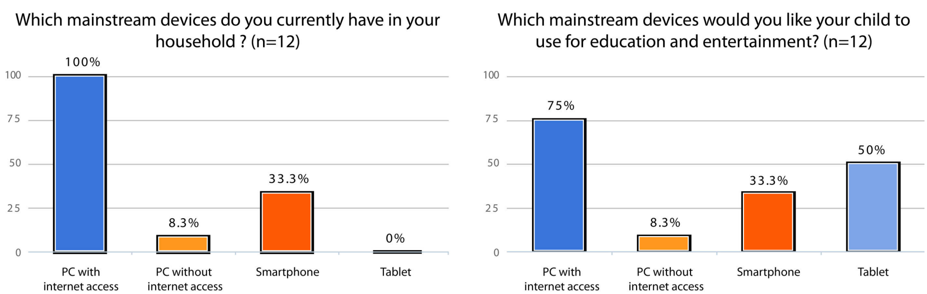


Fig. 4. Mainstream devices that are available to users the household and parents' wishes regarding mainstream devices used by children AAC users in the future

5 Conclusion

In the world where people are "always connected" via mobile devices and communicate and obtain information online, it is important to help the people with various types of disabilities. The access to these mainstream technologies is fostering their participation in society.

Since the majority of mainstream mobile devices allow adding extra functionality by downloading and installing optional applications, the adding of symbol-based AAC programs and services in that form is a way to expand the accessibility for some users without hindering user experience for others.

The integration of AAC functionality using applications is a step in the right direction since it helps people who use symbols for communication to stay in touch with the latest technological achievements in mobile technology. These AAC solutions should lend itself to the user, but also to the device and its technical characteristics.

The survey of the habits of Croatian children with disabilities shows that there are numerous problems with the existing AAC programs and services, and that parents want their children to use AAC programs more and for wider variety of tasks. There is also a strong wish that the children use mainstream tablet devices.

Current mainstream mobile devices, especially tablets based on Android and iOS operating systems, have technical capabilities of implementing the proposed model of adaptable symbol-based AAC system. So as to develop the system and define its graphical and functional aspects, a further research is necessary. Model prototypes can be used to enable the definition and evaluation of these aspects by potential users.

Acknowledgements. This paper is the result of a scientific research within the development project "ICT systems for Persons with Complex Communication Needs", supported by the Development Fund of the University of Zagreb.

References

1. State of the Media: Mobile Media Report Q3 2011. Nielsen U.S., New York (2011)
2. Abascal, J., Nicolle, C.: Moving towards inclusive design guidelines for socially and ethically aware HCI. Interacting with Computers 17, 484–505 (2005)
3. Williams, M.B., Krezman, C., McNaughton, D.: Reach for the Stars: Five Principles for the Next 25 Years of AAC. Augmentative and Alternative Communication 24(3), 194–206 (2008)
4. eAccessibility – Opening up the Information Society. European Commision (2010), http://ec.europa.eu/information_society/activities/einclusion/policy/accessibility/index_en.htm
5. Glennen, S.L., DeCoste, D.C.: The handbook of augmentative and alternative communication. Singular, San Diego (1997)
6. DeRuyter, F., McNaughton, D., Caves, K., Bryen, D.N., Williams, M.B.: Enhancing AAC connections with the world. Augmentative and Alternative Communication 23(3), 258–270 (2007)
7. Curry, D.G., Martinsen, G., Hopper, D.G.: Capability of the human visual system. In: Hopper, D.G. (ed.) Cockpit Displays X. Proceedings of SPIE, vol. 5080, pp. 58–69. SPIE, Bellingham (2003)
8. Wright, S.L.: IBM 9.2-megapixel flat-panel display: Technology and infrastructure. In: Hopper, D.G. (ed.) Cockpit Displays IX: Displays for Defense Applications. Proceedings of SPIE, vol. 4712, pp. 24–34. SPIE, Bellingham (2002)

9. Gille, J., Samadani, R., Martin, R., Larimer, J.: A Human Visual Discrimination Model Analysis of the Gray Scale/Resolution Tradeoff: Displays with 150 dpi or Less Resolution. In: Proceedings of the Society for Information Display, pp. 494–497. Society for Information Display (1994)
10. Larimer, J., Gille, J., Luszcz, J.: Display Resolution Tradeoffs: Grayscale Spacing. In: Conference Record of the 1997 International Display Research Conference and International Workshops on LCD Technology and Emissive Technology, pp. 262–265 (1997)
11. Sheedy, J.E.: Vision Problems at Computers. In: Handbook of Optics. Optical Society of America, vol. III. McGraw-Hill, New York (2001)
12. State of the Media: Mobile Media Report Q3 2011. Nielsen U.S., New York (2011)
13. Cozza, R., Milanesi, C., Zimmermann, A., Glenn, D., Gupta, A., De La Vergne, H.J., Lu, C.K., Sato, A., Nguyen, T.H., Shen, S.: Market Share: Mobile Communication Devices by Region and Country, 3Q11. Gartner (2011)
14. Mobile devices & communication apps; An AAC-RERC White Paper. AAC-RERC (2011), http://www.aac-rerc.com/
15. Car, Z., Vukovic, M., Vucak, I., Pibernik, J., Dolic, J.: A Platform Model for Symbol Based Communication Services. In: Plank, T., Löschnigg, M. (eds.) Proceedings of the 11th International Conference on Telecommunications ConTEL 2011, pp. 141–147. Graz University of Technology, Graz (2011)
16. Schulz, T., Fuglerud, K.: Universal Mobile Device (UMD) - Methods, Inventory, and Dissemination. Norwegian Computing Center, Oslo (2010)

Potentials of the Tangible User Interface (TUI) in Enhancing Inclusion of People with Special Needs in the ICT-Assisted Learning and e-Accessibility

Matej Zajc[1] and Andreja Istenic Starcic[2,3]

[1] University of Ljubljana, Faculty of Electrical Engineering, Ljubljana, Slovenia
matej.zajc@fe.uni-lj.si
[2] University of Ljubljana, Faculty of Civil and Geodetic Engineering, Ljubljana
[3] University of Primorska, Faculty of Education, Koper, Slovenia
andreja.starcic@siol.net

Abstract. In the modern society, inclusion of persons with disabilities in the mainstream education is increasingly enhanced. The presented example focuses on inclusion of individuals for whom their learning environment is created and adapted according to their special needs and capabilities. In establishing the learning environment, the role of multimedia and ICT is important. Persons with disabilities are often disadvantaged in ICT and in participating in the classroom collaborative activities. The paper presents a novel interactive application utilizing tangible user interface (TUI) as an alternative input device for computer-supported collaborative learning and game based-learning. In the final part of our paper, we summarize the main findings and results of evaluation of our interactive game for the ICT-assisted learning of people with special needs often having them tested in a real educational environment. The findings indicate high usability of TUI for people with physical impairment and severe to mild learning difficulties for whom it enables inclusive ICT and e-accessibility.

Keywords: tangible user interface, TUI, assisted learning, ICT, multimedia, e-accessibility, inclusion.

1 Introduction

Introduction of innovative and intuitive user interface systems can improve learning and teaching of diverse learner groups by enabling an easier computer-human interaction and as a result ubiquitously introducing end users, i.e. learners, to computers and multimedia.

The aim of our previous work was to utilize tangible objects in learning to follow the procedure on the paper for addition concepts [11] and geometry concepts [13] with the user actions being analogous to the real world effects or similar actions [6].

This paper presents a novel research effort in investigating the role of a tabletop system using tangible user interface (TUI) in the classroom environment and in developing innovative interactive game.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 261–270, 2012.

The system to be used by the target-group consisting of diverse groups of learners with special needs should be intuitive and user-friendly. On the other hand, its developers should be offered a simple environment to develop applications with support of developers community. The tasks pursued in our research were to:

1. enhance computer-supported collaborative learning and interaction for people with special needs and for diverse learner groups in inclusive classrooms, and
2. facilitate game-based learning supported by an innovative approach to learning by playing through interacting with tangible objects.

2 Pedagogical Approaches to Inclusion in the Mainstreaming Education for Persons with Special Needs

Efforts to allow for equal educational opportunities for persons with disabilities and their integration into mainstream programs require new approaches and strategies to teaching and new learning methods. Before the onset of inclusion, persons with disabilities were expected to integrate with the mainstream classes; the prevailing tendency was to adapt of individuals to the environment.

The inclusion approach does quite the opposite. It puts the individual in focus and adapts the environment to individual needs and abilities. Emergence of the inclusive classes in promoting inclusion in education is very important, but also very difficult in terms of teaching approaches, strategies and learning methods. The most important among the key competencies of the teachers for the day-to-day teaching in the inclusive classroom are particularly differentiation and individualization, identification of progress and knowledge of individuals and identification of individual approaches and styles of learning integrated into the learning environment.

The role of ICT and educational technology in the development and support of teaching approaches, strategies and learning methods is important. The potential of ICT in the field of special educational needs is not yet fully exploited, most of the hardware and software and computer interfaces have been developed for the mainstream population [10], [33]. Therefore, for the people with special needs using ICT can pose a significant obstacle. In the last decade e-accessibility has been dedicated much attention. Important developments and projects of the World Wide Web Consortium - W3C [31] and the Web Accessibility Initiative - WAI [32] have raised the degree of accessibility for different groups of people with special needs such as people with auditory, cognitive, neurological, physical, speech, and visual impairments. The assistive technology for specific areas of disabilities is highly developed, while the ICT-supported learning for people with special needs has in the past received less attention [10]. In the period between December 2010 and January 2012, an extensive review study was conducted, together with a qualitative analysis of the published scientific articles indexed in the SSCI journals during the period from 1970 to 2011. Its findings show a very small proportion of publications dealing with the ICT-supported learning of persons with disabilities compared to the ICT-supported learning for the mainstream population. Among the collected articles there were only 10% of those addressing inclusion in education enabled by ICT-based learning of persons with special needs. The goals of one of the key projects funded by the EU Lifelong Learning program, adopted in the EU, Australia, Korea and India to eliminate this deficiency, i.e. to enable ICT-supported learning for people having special needs [4], was to raise awareness and to develop competences of the end users.

One of the most important elements of ICT is the human-computer interface. Its diversity, which combine different cognitive and sensory-motor functions, is very important in providing accessibility to different groups of users. TUI supports people with different disabilities enabling inclusive ICT and e-accessibility.

Our study was conducted in two groups: an inclusive classroom with learners with physical impairment and learning difficulties; a classroom in vocational education for students with severe learning difficulties. Students with different levels of motor impairment need adjustments for being able to use computers. The input devices, i.e. the keyboard in mouse, are in many cases inadequate; adjustments are required to the type of gripping, gripping patterns and patterns of their users. For the students with severe learning difficulties, the human-computer interface has to overcome a number of physical, communicational, emotional and cognitive disabilities causing learning difficulties in reading, writing and perceiving [12].

Among the important impacting factors in providing human-computer interface to broader user groups and their introduction in the educational process is cost effectiveness. In our paper we propose a low cost solution suitable for a widespread use.

3 Computer Interfaces Used in Creation of Learning Environments

Children are already, at the stage of early learning, using digital technologies for playing, utilizing computers, smartphones, TV sets and dedicated hardware. Digital games are increasingly popular for children and adults and have penetrated in learning environments. Game-based learning has proved its potential in facilitating learning at home and in the classroom [2]. At school, multimedia technology penetration in the classrooms is somehow limited to computer rooms and interactive whiteboards with projectors.

Over 30 years after its invention, the computer mouse is still in use as the input device enabling wide possibilities of human interaction with the computer [19]. The computer mouse is used to access the multiple graphical user interface (GUI) controls in sequence [7], [29] where a single device is used to complete the required actions. As a consequence, the mouse interaction requires the user to continuously couple and decouple digital objects in the problem-solving process [26].

For a novice user, as children for example, it is quite difficult to efficiently interact with the application using the mouse [8]. In most cases it requires adaptation of motoric skills for specific problem solving.

An alternative input present tabletop systems broadly categorized into touch and tangible technologies [21]. Recently, the touch technologies are becoming dominant with smartphones and tablet computers. Multi-touch systems, offered by tablet computers for example, offer several advantages over TUIs, providing a more dynamic content and control capabilities.

Nevertheless, tangible systems provide some advantages for diverse learner interactions in the learning process investigated in this paper. Moreover, the research conducted by [29] reveals that the control tasks with TUIs are quicker to acquire and more accurate to manipulate compared to the multi-touch input methods.

TUIs [9], [24-25] present an interesting solution as an alternative user interface where physical objects enable the user to grasp and to manipulate thus making it especially suitable for young children [20], [34]. Children are used to handle physical objects in a natural manner [26]. With an appropriate application, TUIs enable users to perform goal-oriented activities and also trial-and-error activities [26].

TUIs enable an innovative, conceptually different interaction with the computer. Physical manipulation on a tabletop is transformed into computer commands and displayed in the application domain on the computer screen. In addition, TUIs enable two-handed manipulation which is favorable from the cognitive and motoric perspective [17]. Tangible technologies are already well accepted in the learning domain [20], [24-25].

Recent projects offer open-source tool-kits enabling development of dedicated TUIs [16] targeting at edutainment [3], [5], [14], and [15]. Among different solutions a robust, cost-effective and open-source solution [28] was selected as a platform for designing an interactive application for learning purposes. With the Trackmate platform, the authors follow the ‘low floor-wide walls’ paradigm [23], allowing users an easy start and enabling developers to design a wide range of applications, as for example [11] and [13].

3.1 Classroom Environment

The table is a natural school environment for learning, collaboration and problem solving. The idea to introduce tabletop systems seems natural and enables opportunity to ubiquitously introduce a computer in the classroom. The desktop system is not limited to a single user since children can sit around the interactive table and collaborate with their friends, thus providing a real social experience that facilitates learning [13]. In [13] authors report on different social modes in classroom learning enabled by a desktop system with TUI: teacher up front, individual, small group and pair.

TUI presents a natural combination of physical and digital worlds exploring advantages of natural interfaces that are intuitive and easy to learn [25]. In learning, users need to track the state of their activity to monitor the progress of their work [26]. Physical manipulatives are useful to represent abstract concepts with physical representations. To assure positive results, the teacher’s guidance is of vital importance. This arguments support the use of such systems in small dedicated groups.

The proposed system utilizes a traditional computer display. With the recent developments in the LCD technologies, it has become a high resolution and low-cost device. This is in line with our objective to use the equipment available in the classroom.

4 Method

A case study was conducted to support the application development and to identify the use of TUI for groups of end users. Evaluation of the final version of the Raindrop catcher, presented in 5.1, was tested in 2010 with two groups of end users. Our case study supported investigation of the TUI use in an authentic educational setting. The final evaluation testing was conducted with two groups. One group of participants was formed of primary school pupils aged 7-8 years. Altogether 23 students, 11 girls and 12 boys participated in our study. One group of students was formed of 14 vocational-school students aged 16-17 years, 6 girls and 8 boys. The data was collected by teachers and experts observing students working individually and in groups. The focus of our final evaluation testing was on two questions: (1) Was the computer-supported collaborative learning well applied in the inclusive classroom and for TUI for students having difficulties in using the computer's input devices: keyboard and mouse? (2) Was the TUI-supported game-based learning efficient for the students with motor impairment and for the students with severe to mild learning difficulties?

5 System Architecture

The Trackmate open-source system was selected as a reference system in designing a prototype system meeting our requirements for an open-source initiative to create an inexpensive system [28]. It is based on a software environment allowing computers to recognize the tagged objects when placed on a tabletop surface. It consists of a tangible tabletop system connected to a computer.

After designing the initial prototype of the tabletop system, we turned our ideas into practical applications. The proposed tabletop system is composed of TUI connected to a computer running applications that are displayed on a monitor. The proposed system presented in [11] is portable, robust and compact.

The main component of this tangible tabletop system is a transparent surface where tangible objects are placed. The camera, placed below, continuously captures positions and rotations of each of these objects. Each object has a unique circular barcode. The object is detected by the camera and decoded by the Trackmate Tracker [16], which sends the object data via LusidOSC, a protocol layer for unique spatial input devices, to LusidOSC-based applications [18]. The interactive application is programmed in Processing [22]. The information about the user interactions with objects is used as an input data in the application.

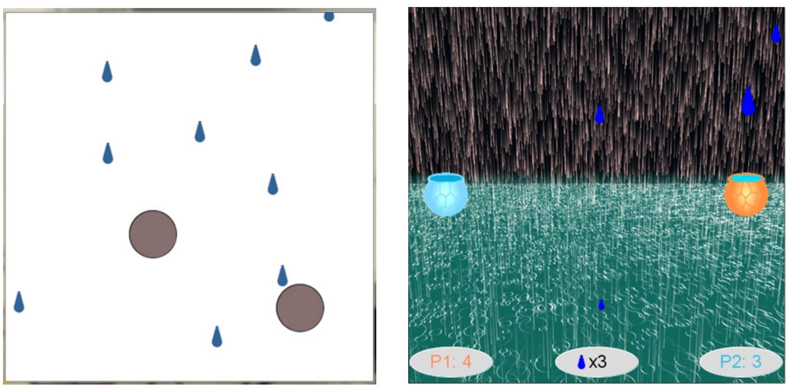
5.1 Interactive Game Design and Development - Raindrop Catcher

In this paper the application Raindrop catcher is used as a case study to evaluate potentials of TUI in enhancing inclusion of people with special needs.

The presented work was conducted in collaboration with an interdisciplinary team of experts from different fields: electrical engineering, education, didactics and special didactics. Based on the shared ideas, the Raindrop catcher was developed at weekly intervals from initial design (Fig. 1a) to the final version (Fig. 1b) which was tested on two groups of students in 2010. The first version was used to test how students interact with the system. In the initial stage the game was designed from scratch and with the positive feedback upgraded to the final version.

The Raindrop catcher (Fig. 2) is a simple game where the player catches raindrops displayed on the monitor by moving the tangible object on the tabletop. The game allows each of the two users playing simultaneously on the same tabletop handle the corresponding object, i. e. the catcher. The goal is to reach as many points where the raindrops of different sizes result in a different number of points. The game ends when one if the users misses a preset number of raindrops.

Each tangible object has a corresponding catcher on the monitor. The application continuously retrieves information for each tangible object's id and position from the Trackmate Tracker [16] via the LusidOSC protocol [18] and properly positions catcher on the monitor. The raindrops are randomly dropping from the top of the monitor. The application monitors if the catcher properly interacts with the raindrops and updates the results.



a) b)

Fig. 1. Raindrop catcher – TUI application used in the case study: a) first version, b) final version

Fig. 2. provides more details about the concept of the Raindrop catcher. The application tracks the tangible objects located on the tabletop and displays the action on the monitor (Fig. 2.1). The objective is to catch the raindrops by appropriate movements of the objects on the tabletop surface. The game allows two players to play simultaneously (Fig. 2.2). Each player collects points by collecting the raindrops with a catcher (Fig. 2.3). The raindrop's size defines the number of points to be collected. While the level of the collected points increases the size of the catcher decreases (Fig. 2.4) and the time intervals between the raindrops get shorter (Fig. 2.5). The game ends when one of the players misses the pre-set number of the raindrops (Fig. 2.6).

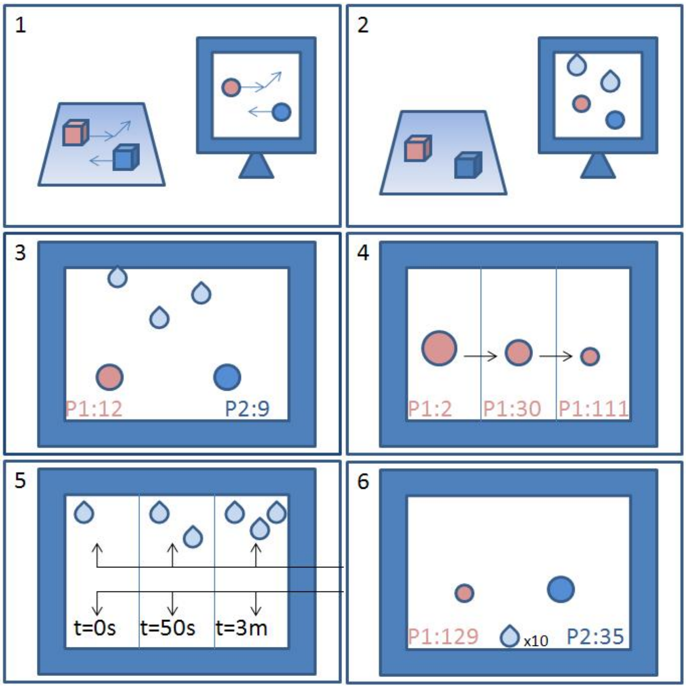


Fig. 2. Concept of the Raindrop catcher

6 Findings and Discussion

The study was focused on two research questions to identify usefulness of computer supported by TUI in inclusive classroom and efficiency of TUI-supported game-based learning for the students with motor impairment and for the students with severe to mild learning difficulties. These students are usually deprived for computer gaming with classical user interfaces. TUI is useful for students who find it difficult to use the computer by the input devices, i.e. the keyboard and the mouse. Results of our evaluation testing prove that TUI enabled these students for equal engagement in the game-based learning process.

In addition, the use of TUI improves the collaboration space, thus supporting two persons in using the computer synchronously, by exchanging their left and right hand. In a lecture room, TUI in its role of an interactive tabletop system, allows groups of students to move around an interactive table while exchanging themselves and interacting. In the evaluation it was observed that students' motivation increased with improvements of graphical design of the game (Fig. 1).

The computer use is equally engaging with regard to cognitive and motor engagement of both the mainstream learners and those with learning difficulties and motor impairment. For the physically disabled students, the interactive table is reachable from their wheel-chair. Our conclusion is that TUI efficiently supports collaborative learning in the inclusive classroom.

Computer games form an important learning environment at all levels of education and for all age groups. Game-based learning supported by TUI enables inclusion of students with motor impairment and students with severe to mild learning difficulties and results in improved learning process. Students having difficulties using keyboard and mouse were actively engaged in the learning process with lively group discussions.

7 Conclusion

Education can significantly benefit from TUI by enabling learners to work with actual physical objects supplemented with computing power. Evaluation findings of the presented interactive game, together with applications presented in [11] and [13], prove Trackmate system [28] to be a very useful and low-cost interactive interface for the school environment.

TUI supported game-based learning enabled students with motor impairment and students with severe to mild learning difficulties to overcome usual limitations with using computers.

An inexpensive hardware and robust application make it an ideal solution for being used in the classroom environment. These attributes together with the Trackmate as an open-source environment are found attractive also for other members of the developers' community.

Further work is planned in the area of tabletop design following user-centered design approach to satisfy different user-group needs. Users collaboration can be improved by extending the environment to be internet enabled and consequently offering improved tutoring and monitoring capabilities.

References

1. Apitz, G., Guimbretière, F., Zhai, S.: Foundations for designing and evaluating user interfaces based on the crossing paradigm. ACM Transactions on Computer-Human Interaction 17(2), 1–42 (2010)
2. Bober, M.: Games Based Experiences for Learning, `http://www.futurelab.org.uk/sites/default/files/Games_based_experiences_for_learning_0.pdf`
3. Costanza, E., Shelley, S.B., Robinson, J.: Introducing audio d-touch: A tangible user interface for music composition and performance. In: Davies, M. (ed.) Proceedings of the 2003 International Conference DAFx 2003 (2003)
4. ENABLE: ICT supported learning for people with special needs, `http://i-enable.eu`
5. Feltham, F.G.: Do the blocks rock: A tangible interface for play and exploration. In: Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat, pp. 188–194. ACM, New York (2008)
6. Fishkin, K.: A taxonomy for and analysis of tangible interfaces. Personal and Ubiquitous Computing 8(5), 347–358 (2004)

7. Fitzmaurice, G.W., Buxton, W.: An Empirical Evaluation of Graspable User Interfaces: towards specialized, space-multiplexed input. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 43–50. ACM, Atlanta (1997)
8. Inkpen, K.M.: Drag-and-drop versus point-and-click mouse interaction styles for children. ACM Transactions on Computer-Human Interaction 8(11), 1–33 (2001)
9. Ishii, H.: The tangible user interface and its evolution. Communications of the ACM 51(6), 32–36 (2008)
10. Istenic Starcic, A.: Educational technology for inclusive classroom. TOJET 9(3), 26–37 (2010)
11. Istenic Starcic, A., Zajc, M.: An interactive tangible user interface application for learning addition concepts. British Journal of Educational Technology 42(6), 131–135 (2011)
12. Istenic Starcic, A., Niskala, M.: Vocational students with severe learning difficulties learning on the Internet. British Journal of Educational Technology 41(6), E155–E159 (2010)
13. Istenic Starcic, A., Cotic, M., Zajc, M.: Design-based research on the use of a tangible user interface for geometry teaching in an inclusive class-room. British Journal of Educational Technology (accepted for publication, 2012)
14. Jordà, S., Geiger, G., Alonso, M., Kaltenbrunner, M.: The reacTable: Exploring the synergy between live music performance and desktop tangible interfaces. In: Proceedings of the 1st International Conference on Tangible and Embedded Interaction, TEI, pp. 139–146. ACM, New York (2007)
15. Juan, C., Canu, R., Giménez, M.: Augmented reality interactive storytelling systems using tangible cubes for edutainment. In: Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies, ICALT, pp. 233–235. IEEE Computer Society, Los Alamitos (2008)
16. Kumpf, A.: Trackmate: Large-scale accessibility of tangible user interfaces (unpublished master's thesis), Massachusetts Institute of Technology, Cambridge, MA (2009)
17. Leganchuk, A., Zhai, S., Buxton, W.: Manual and cognitive benefits of two-handed input: an experimental study. ACM Transactions on Computer-Human Interaction 5(4), 326–359 (1998)
18. LusidOSC, `http://lusidosc.sourceforge.net/`
19. Moggridge, B.: Designing Interactions. MIT Press (2007)
20. O'Malley, C., Fraser, D.S.: Literature review in learning with tangible technologies (Report 12). Nesta Futurelab Series, Bristol (2004)
21. Porta, M.: Human–computer input and output techniques: an analysis of current research and promising applications. Artificial Intelligence Review 28(3), 197–226 (2009)
22. Processing.org, `http://processing.org/`
23. Resnick, M., Silverman, B.: Some reflections on designing construction kits for kids. In: Eisenberg, M., Eisenberg, A. (eds.) Proceedings of the 2005 Conference on Interaction Design and Children, pp. 117–122. ACM, New York (2005)
24. Shaer, O., Hornecker, E.: Tangible user interfaces: Past, present, and future directions. Foundations and Trends in Human–Computer Interaction 3(1-2), 1–137 (2009)
25. Shaer, O., Jacob, R.J.K.: A specification paradigm for the design and implementation of tangible user interfaces. ACM Transactions on Computer-Human Interaction 16(4), 1–39 (2009)
26. Sharlin, E., Watson, B., Kitamura, Y., Kishino, F., Itoh, Y.: On tangible user interfaces, humans and spatiality. Personal and Ubiquitous Computing 8(5), 338–346 (2004)

27. Toney, A., Thomas, B.H.: Considering Reach in Tangible and Table Top Design. In: Proceedings of the 1st IEEE International Workshop on Horizontal Interactive Human-Computer Systems, pp. 57–58. IEEE CS, Adelaide (2006)
28. Trackmate, `http://trackmate.sourceforge.net/`
29. Tuddenham, P.: Graspables Revisited: Multi-Touch vs. Tangible Input for Desktop Displays in Acquisition and Manipulation Tasks. In: Sarcevic, A. (ed.) Proceedings of the 28th International Conference on Human Factors in Computing Systems, pp. 2223–2232. ACM, New York (2010)
30. Wong, A.W.K., Chan, C.C.H., Li-Tsang, C.W.P., Lam, C.S.: Competence of people with intellectual disabilities on using human–computer interface. Research in Developmental Disabilities 30, 107–123 (2009)
31. W3C Leading the Web to its full potential, `http://www.w3c.org`
32. W3C WAI Web accessibility initiative, `http://www.w3c.org/WAI`
33. Wong, A.W.K., Chan, C.C.H., Li-Tsang, C.W.P., Lam, C.S.: Competence of people with intellectual disabilities on using human–computer in-terface. Research in Developmental Disabilities 30, 107–123 (2009)
34. Xu, D., Read, J.C., Mazzone, E., MacFarlane, S., Brown, M.: Evaluation of Tangible User Interfaces (TUIs) for and with Children – Methods and Challenges. In: Jacko, J.A. (ed.) HCI 2007. LNCS, vol. 4551, pp. 1008–1017. Springer, Heidelberg (2007)

Rock Art Interpretation within Indiana MAS

Viviana Mascardi[1], Vincenzo Deufemia[2], Damiano Malafronte[1], Alessandro Ricciarelli[1], Nicoletta Bianchi[3], and Henry de Lumley[3]

[1] Università di Genova
Via Dodecaneso 35, 16146, Genova, Italy
viviana.mascardi@unige.it,
{damianomal,ricciarelli.alessandro}@gmail.com

[2] Università di Salerno
Via Ponte don Melillo, 84084 Fisciano (SA), Italy
deufemia@unisa.it

[3] Lab. Dép. de Préhistoire du Lazaret
33 Bis Boulevard Franck Pilatte, 06300 Nice, France
{nbianchi,lazaret}@lazaret.unice.fr

Abstract. This paper presents the first results achieved within the Indiana MAS project funded by Italian Ministry for Education, University and Research, MIUR. We discuss how the AgentSketch holon belonging to the Indiana MAS has been extended to cope with images, besides hand drawn sketches, and has been tested in the domain of Mount Bego's prehistoric rock art (southern French Alps). The way Indiana MAS holons cooperate in order to provide correct interpretations of ambiguous shapes is discussed by means of an example based on hypotheses recently advanced by archaeologists.

1 Introduction

This paper presents the first results achieved within the project "Indiana MAS and the Digital Preservation of Rock Carvings: A Multi-Agent System for Drawing and Natural Language Understanding Aimed at Preserving Rock Carvings", funded by Italian Ministry for Education, University and Research, MIUR. The project, accepted for funding in September 2011, will start in March 2012.

The aim of the project is to develop a technology platform based on intelligent software agents for the digital preservation of rock carvings, which both integrates and complements the techniques usually adopted to preserve heritage sites. The platform will support domain experts in the creation of a repository, which may become a reference at Italian and European level as a thorough database of rock carvings, and in the interpretation of rock carvings. It will also promote the awareness and the preservation of the cultural treasure by making cultural information accessible to all on the Internet and preserve it for future generations. To this end, the Indiana MAS platform will enable the preservation of all kinds of available data about rock carvings, such as images, geographical objects, textual descriptions of the represented subjects. It will provide the means to organize and structure such data into an existing collaborative tool

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 271–281, 2012.

set, and will supply domain experts with collaborative facilities for processing data and making assumptions about the way of life of the ancient people based on them.

The choice of agent technology for addressing Indiana MAS goals was a very natural one, given the need that each component of the system, while operating in a highly autonomous way, interacts and coordinates with the other components to share information and to reason about them in the most effective way.

The paper is organized in the following way: Section 2 introduces the Indiana MAS architecture and provides some background information on AgentSketch. Section 3 discusses how the AgentSketch component of Indiana MAS has been extended to cope with images, besides hand drawn sketches, and has been tested in the domain of prehistoric rock art, and illustrates how correct interpretation can result from interaction between AgentSketch and other MAS components. Section 4 concludes and outlines our future work.

2 Indiana MAS and AgentSketch

Indiana MAS integrates intelligent software agents, ontologies, natural language processing and sketch recognition techniques. Multi-agent systems (MASs) represent an optimal solution to manage and organize data from multiple sources and to orchestrate the interaction among the components devoted to the interpretation of the carvings. Ontologies allow to define a common vocabulary that can be profitably exploited to organize data associated with rock carvings, included their semantic annotations, and create semantic relationships between them. Natural Language Processing techniques are used to extract relevant concepts from text and for mining semantic relationships among them, hence supporting the definition and evolution of ontologies devoted to describe the domain. Sketch recognition techniques are applied to classify the elementary shapes of the carving drawings and associate their possible interpretations with them.

Fig. 1 describes the Indiana MAS holonic architecture. Each component represented by a 3D puppet in the figure is a holon: it consists of "agents that give up parts of their autonomy and merge into a super-agent (a holon), that acts - when seen from the outside - just as a single agent again" [8].

Starting from structured and unstructured multilingual and multimedia data coming from different sources (the Bicknell Legacy owned by the University of Genova including images of Bicknell's tracings with metadata in English and in Italian; the ADEVREPAM database owned by the Laboratoire Départementale de Préhistoire du Lazaret storing images of the tracings made by De Lumley's team with metadata in English and French; heterogeneous data coming from other open sources such as the World Wide Web), we created a first repository of raw data.

AgentText takes care of understanding text in English, French, and Italian, and classifying it according to the Rock Carving Vocabulary that dynamically evolves as new data will be inserted into the repository.

Image and drawing understanding are performed by AgentSketch [2], a multi-agent system that we developed as part of our previous research, and that will

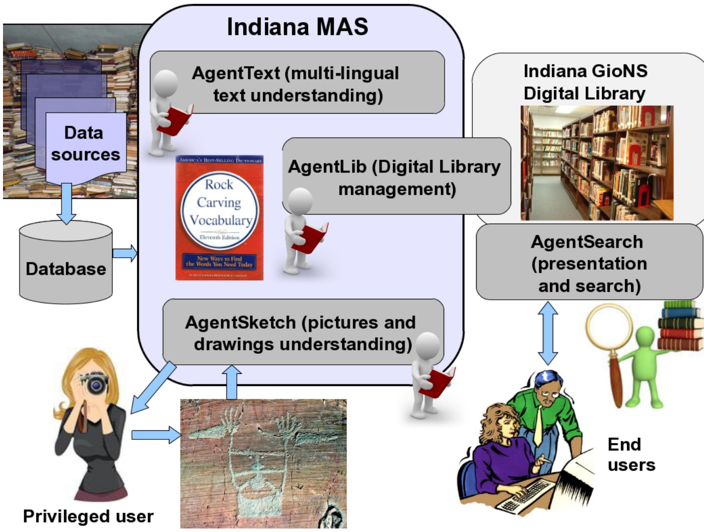


Fig. 1. Indiana MAS architecture

be discussed in the sequel. Classified data (text, images, drawings) are made available to the public thanks to the Indiana GioNS Digital Library. The Library is managed by AgentLib, that interacts with both AgentText and AgentSketch to help relationships among heterogeneous data emerge and be stored into the Digital Library as new knowledge. The Library is made accessible to users by AgentSearch, a personal user assistant designed along the lines of old and well-know "digital butlers" [4,10], providing user profiling and content personalization capabilities. The interface offered by AgentSearch integrates query-by-sketch and natural language understanding technologies.

While the access point to Indiana MAS services for end users is AgentSearch, each holon in the MAS can be directly accessed by privileged users, as shown in Fig. 1 where an archaeologist inputs the image of Mount Bego's most famous engraving, the "wizard", to AgentSketch directly, for directly obtaining an interpretation of the image.

AgentSketch is composed by the four kinds of agents shown in Fig. 2.

Interface Agent. It represents an interface between the agent-based AgentSketch framework and the generic "Input Suppliers" that are not included inside the framework. The nature of these input suppliers may vary according to the type of sketch to be interpreted and to the drawing process (on-line vs. off-line). The Interface Agent informs the "Sketch Interpretation Agent" (SIA) and the "Input Pre-Processing Agent" (that, in turns, informs the "Symbol Recognition Agents") about the nature of the recognition process (off-line or on-line) and converts the information produced by the input suppliers into a suitable format for these agents. It sends each new available piece of input (or the whole input,

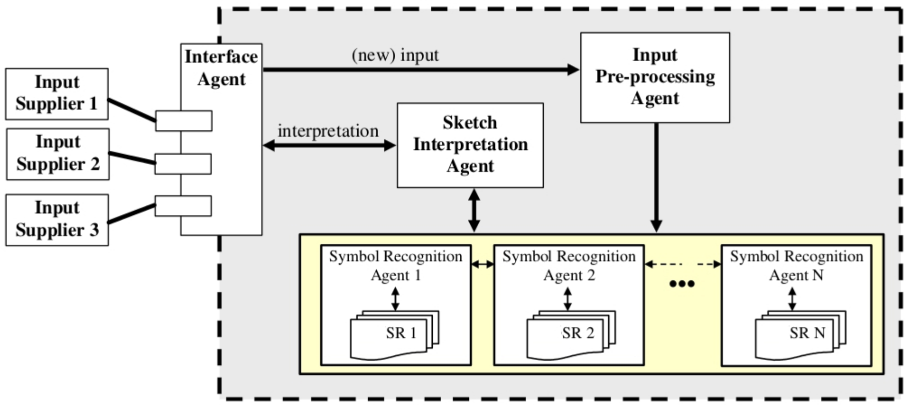


Fig. 2. AgentSketch architecture

in case of an off-line recognition process of an image) to the Input Pre-Processing Agent, and interacts with the SIA for sending the sketch interpretation requests to it, and for delivering its answer to the user. For example, the implemented instance of the Interface Agent currently available within AgentSketch provides a stub for an on-line editor that takes advantage of Java Swing components and Satin, and for a file browser allowing the user to select images from the file system.

Input Pre-processing Agent. It processes the input received from the Interface Agent and sends the obtained results to the "Symbol Recognition Agents" described in the following, using a format compliant with the recognition approach they apply.

Symbol Recognition Agents (SRAs). Each SRA is devoted to recognize a particular symbol of the domain. Moreover SRAs may collaborate with other SRAs in order to apply context knowledge to the symbols they are recognizing, and with the SIA that deals with the sketch interpretation activity. On-line recognition of hand-drawn sketches and off-line recognition of objects within images are based on very different approaches, and exploit different algorithm and existing libraries to work. This lower level activity is demanded to "Symbol Recognizers" (SRs in the figure) that are not agents, lacking most of the agent-characterizing features, but just software modules managed by SRAs. As long as there is one SRA that correctly integrates SRs by managing their execution as well as data conversion issues, the actual implementation of the SRs and the approach to recognition that they adopt do not matter. The inherently flexible and modular agent-based approach allowed us to seamlessly cope with symbols that have been recognized by heterogeneous SRs managed by ad-hoc SRAs.

Sketch Interpretation Agent (SIA). The SIA provides the correct interpretation either of the sketch drawn so far (in case of an on-line drawing process) or of the entire sketch (in case of an off-line recognition process) to the Interface Agent. In particular, it analyzes the information received from SRAs and solves conflicts between symbols that might arise. When all the conflicts have been solved, the SIA proposes the sketch interpretation to the user, interacting with the Interface Agent.

AgentSketch has been experimented in the domain of Use Case Diagrams [2]. Its exploitation for reasoning about hand-drawn sketches in the physical security domain has been discussed in [3]. In the following section we discuss two SRAs operating in the rock art domain, and the interactions among AgentSketch and the other holons in the Indiana MAS to reach an agreement on the correct interpretation of a sketch, based on external information coming from heterogeneous and multi-lingual documents.

3 Recognizing and Interpreting Rock Carvings in Indiana MAS

3.1 SRA for Sketches

In order to recognize rock art carvings while they are being sketched, we implemented a SRA working on-line and based on LADDER [9]. In LADDER, symbol recognition is performed using the rule-based system Jess [7]. In particular, for each symbol of the domain, a Jess rule is automatically generated from a LADDER structural shape description, which mainly contains information on the shape of the symbol. Recognition using Jess is sensible neither to the order of the strokes, nor to the symbol dimension. These features make the approach very stable.

As an example, the LADDER description of a corniculate is shown in Fig. 3. This rule defines a corniculate as one ellipse and four lines from which the shape is built, plus the topological constraints defining the relationships among these elements: two lines (representing the lower part of the horns) must be similar in length, touch the ellipse (representing the head), and be over it; the two other lines (representing the upper part of the horns) must be similar in length, and each of them must be above the lower part of one horn and connected to it.

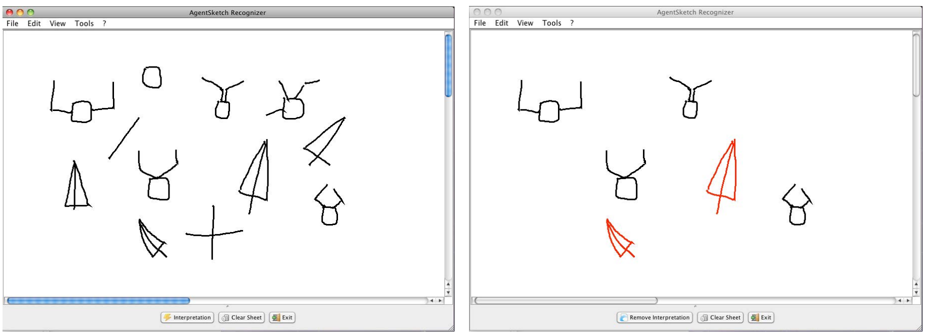
The final behavior of this rule is that, when in the working memory of the LADDER application there are four lines and one ellipse that respect the precondition of the rule, the rule is fired and a corniculate symbol is recognized. A similar (but simpler) rule has been defined for recognizing short weapons.

The shapes described with LADDER must be diagrammatic or iconic since they have to be drawn using a predefined set of primitive shapes and composed using a predefined set of constraints.

Fig. 4 shows a screenshot of AgentSketch at work. Fig. 4(a) shows the symbols sketched by the user, and 4(b) shows which of them have been correctly recognized as corniculates (black symbols) and as daggers (red – or light gray if printed in gray-scale – symbols). All the symbols have been correctly recognized as belonging to their own category, apart from the dagger in the lower left portion

```
(define shape Corniculate
  (description
    "Symbol representing a corniculate in Mount Bego's carvings"
  )
  (components
    (Ellipse head) (Line horn1_lower) (Line horn2_lower) (Line horn1_upper) (Line horn2_upper)
  )
  (constraints
    (touches head horn1_lower) (touches head horn2_lower)
    (hornsConnected horn1_lower horn1_upper) (hornsConnected horn2_lower horn2_upper)
    (equalLength horn1_lower horn2_lower) (equalLength horn1_upper horn2_upper)
    (equalLength horn1_lower horn2_lower)
     (over horn1_lower head) (over horn2_lower head) (over horn1_upper horn1_lower) (over
horn2_upper horn2_lower)
  )
)
```

Fig. 3. LADDER description of a corniculate



a b

Fig. 4. The user interface of the system

of the screen. Surprisingly, despite the simplicity of their shape, correctly recognizing daggers raises some problems. Daggers that are sketched almost in the same way may be either recognized or not, depending on very small variations of the length and relative position of their constituent elements. Relaxing the constraints of the rule for recognizing daggers would eliminate this problem, but would lead to detection of false positives that, in the current setting, are completely absent. We prefer to keep the rule constraints stricter and having some weapons not recognized, rather than lowering the accuracy of the algorithm by correctly recognizing more weapons, but also having more false positives.

3.2 SRA for Images

The SRA for recognizing rock carvings in images operates off-line and has been developed using an existing open source library of programming functions for real time computer vision, OpenCV (Open Source Computer Vision[1]). Among the

[1] http://opencv.willowgarage.com/wiki/. Last accessed: February the 15th, 2012.

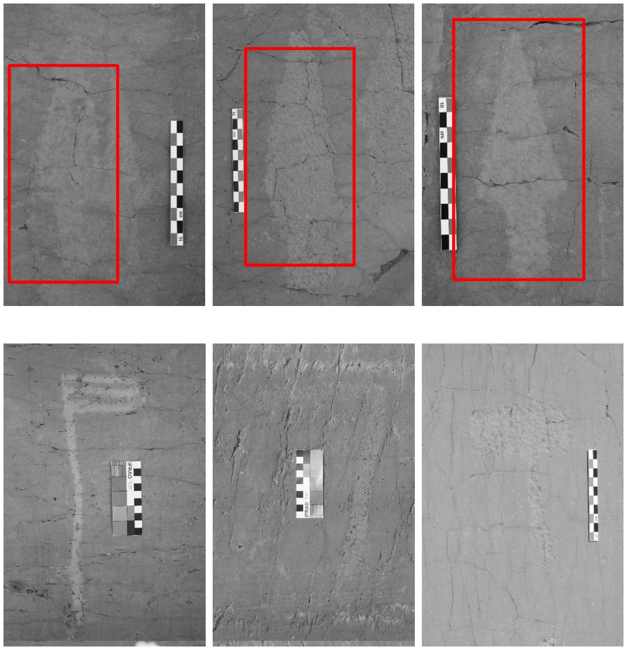


Fig. 5. True positives (top) and true negatives (bottom)

many functions offered by OpenCV, we used those that exploit Haar-like features [14] for categorizing subsections of images based on the intensity of pixels in the region, and the AdaBoost machine learning algorithm [6] for training classifiers to recognize objects in images given positive and negative samples. Some true positives and negatives recognized by the SRA are shown in Fig. 5.

The implementation of another SRA based on SIFT [12] is under way. SIFT is a method for detecting and describing local features in images and has been recently used for automatic coin classification [16]. It could be a good choice both for hand-drawn sketches and for rock carving images, since rock engravings and ancient coins show many similarities from a computer vision viewpoint. In the coin case, SIFT could recognize the same object/element/figure in different coins even if the actual shape varies from one item to another. We are confident to reach the same good results on rock carvings with the SRA we are implementing.

3.3 Interpretation as Cooperation among Holons

Interpreting a given shape within a sketch or an image in the correct way is performed by the standalone AgentSketch (namely, AgentSketch not integrated within Indiana MAS) by exploiting contextual information local to the sketch. For example, in the domain of use case UML diagram, an oval could be interpreted either as the head of a stick man, or as a use case. If there are other strokes in the oval's context (namely, close to it) that follow the pattern of the stick man body, then the oval is more likely to be a head than a use case. In [2] we showed that exploiting such kind of contextual information improves the precision of the interpretation.

When sketches are as complex as rock carvings, however, the context to take into account to provide a correct and precise interpretation cannot be any longer limited to neighboring strokes, and cannot be hard-wired within the AgentSketch Symbol Interpretation Agent.

To make an example, in [5] a systematic analysis of meaningful associations between symbols characterizing Mount Bego's engravings have been carried out, leading to the identification of recurrent patterns such as:

– the dagger, symbol of the light, placed over a reticulate symbolizing the earth; the dagger between the horns of a corniculate (interpretation: the god of storms that fertilizes the earth by means of the rain);
– sinuous lines symbolizing the stream; the canal watering the earth and corniculates associated with sinuous lines and small corniculates; a water basin between the horns of a corniculate (interpretation: water fertilizing the earth);
– a bull below a feminine figure with open arms and legs (interpretation: the myth known as the "son-husband", a symbolic representation of the high goddess or mother goddess giving birth to the bull, who in turn fertilizes the goddess).

Hard-wiring all such patterns within AgentSketch in order to reinforce, for example, the hypothesis that a given shape is a bull if it is below a feminine shape, is not feasible: relationships among symbols are too many, too complex and too dynamic, due to the many co-existing hypotheses made by archaeologists, and to the new ones that are continuously proposed.

The solution implemented in Indiana MAS decouples the discovery and representation of such patterns among symbols from their exploitation during the sketch interpretation stage. The protocol followed by holons to boost interpretation is the following.

1. AgentText is fed with multi-lingual textual documents that it processes to perform the following operations:
   (a) Categorize documents according to the Rock Carving Vocabulary, defined a priory by the domain experts at the set up of Indiana MAS; this activity is carried out by exploiting the techniques discussed in [11]. The output of the categorization activity is sent to AgentLib which stores the documents in the digital library and adds meaningful metadata to them, consistent with the categories discovered by AgentText.
   (b) Extract information about patterns like those described above; this activity is carried out by exploiting the Role Ontology Extractor [1] that we experimented in the archeological domain, as described in [13]. Discovered patterns are stored in the Rock Carving Vocabulary which is implemented as a OWL ontology [15], and hence allows us to easily define relationships among concepts in form of OWL properties that connect them (for example, concepts Dagger and Horn can be connected by the

"between" property, and concepts Dagger and Reticulate by the "over" property, suitably annotated by a pointer to the document where the relationship was extracted from).

Since the success of pattern extraction heavily depends on natural language processing and word sense disambiguation, that are known to be AI-complete problems, domain experts are required to validate patterns proposed by AgentText before they are stored within the Rock Carving Vocabulary. Although not trivial, classification of documents is a more stable task than pattern extraction, and no supervision is required on it.

2. AgentSketch accesses the Rock Carving Vocabulary looking for stored patterns involving symbols that have been recognized so far, whenever it needs to disambiguate the interpretation of a sketch. Patterns that support a given interpretation, make that interpretation stronger. The strongest interpretation is proposed to the user.

This way, patterns are (semi-)automatically extracted from textual documents and represented in a declarative way. No matter what these patters are, where do they originate from, and how fast they evolve during time: AgentSketch always operates in a way that is independent from the values of the patterns themselves, and the resulting interpretation depends on the current set of patterns stored in the Rock Carving Vocabulary.

Fig. 6. Mount Bego's engravings ZIV.GII.R19C.no 12 and 13

To make an example, let us suppose that a sketch like the one represented in Fig. 6 (Mount Bego's engravings ZIV.GII.R19C.no 12 and 13 [5]) is input to AgentSketch. The Symbol Recognizer devoted to recognizing corniculates succeeds in detecting a corniculate in the bottom part of the sketch with a high confidence, but the Symbol Recognizer that should recognize daggers fails in its purpose: the shape in the sketch does not meet the standard dagger shape, and the confidence that could be a dagger is below the usually acceptable threshold. On the other hand, the confidence that it is a sinuous line or a water basin, as computed by the Symbol Recognizers in charge for these two kinds of shapes, is even lower than the confidence that it is a dagger.

AgentSketch looks for patterns involving corniculates in the Rock Carving Vocabulary, and finds the following ones:

– WaterBasin **between** Horns **belongingTo** Corniculate
– Dagger **between** Horns **belongingTo** Corniculate
– Corniculates **closeTo** SinuousLines

These patterns make the interpretation of the unknown shape as a WaterBasin, a Dagger, or a SinuousLine feasible, but since *confidence(Dagger)* > *confidence(WaterBasin)* and *confidence(Dagger)* > *confidence(SinuousLine)*, the interpretation proposed to the user is that the sketch represents a dagger between the horns of a corniculate.

Since each pattern in the Rock Carving Vocabulary contains meta-data about the document it was extracted from, the user is also shown the document as a witness for the given interpretation.

4 Conclusions and Future Work

The digital preservation, classification, and interpretation of rock carvings raises many scientific challenges, such as the integration of data coming from multiple sources, and the interpretation of drawings whose meaning may vary based on contextual information. The Indiana MAS project tackles such issues by exploiting intelligent agents that ensure the required degree of flexibility and autonomy in a highly dynamic and heterogeneous environment. Even if the project did not even start yet, many components of the Indiana MAS have already been developed and tested on the project's domain, and their integration in the Indiana MAS is under way. The steps moved in the rock art interpretation direction discussed in this paper, although preliminary, are very promising. The full achievement of all the project's objectives will be our main goal in the next three years.

Acknowledgments. This research is supported by the "Indiana MAS and the Digital Preservation of Rock Carvings" FIRB project funded by the Italian Ministry for Education, University and Research, under grant RBFR10PEIT.

The authors are grateful to Francesca Odone for her helpful comments on Section 3.2.

References

1. Bozzano, M., Locoro, A., Martelli, M., Mascardi, V.: Trattamento del linguaggio naturale tramite Prolog: un approccio promettente per generare istituzioni virtuali da testi scritti. In: CILC 2010 (2010) (in Italian)
2. Casella, G., Deufemia, V., Mascardi, V., Costagliola, G., Martelli, M.: An agent-based framework for sketched symbol interpretation. J. Vis. Lang. Comput. 19(2), 225–257 (2008)
3. Casella, G., Deufemia, V., Mascardi, V., Martelli, M., Tortora, G.: Reasoning about Hand-Drawn Sketches: An Approach Based on Intelligent Software Agents. In: Sebillo, M., Vitiello, G., Schaefer, G. (eds.) VISUAL 2008. LNCS, vol. 5188, pp. 302–314. Springer, Heidelberg (2008)

4. Chen, L., Sycara, K.P.: Webmate: A personal agent for browsing and searching. In: Agents, pp. 132–139 (1998)
5. de Lumley, H., Echassoux, A.: The Rock Carvings of the Chalcolithic and Ancient Bronze Age from the Mont Bego Area. The Cosmogonic Myths of the Early Metallurgic Settlers in the Southern Alps. L'Anthropologie 113(5P2), 969–1004 (2009)
6. Freund, Y., Schapire, R.E.: A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. In: Vitányi, P.M.B. (ed.) EuroCOLT 1995. LNCS, vol. 904, pp. 23–37. Springer, Heidelberg (1995)
7. Friedman-Hill, E.: Jess, the rule engine for the Java platform. Sandia National Laboratories, Livermore, CA, http://www.jessrules.com/ (last accessed: February 15, 2012)
8. Gerber, C., Siekmann, J.H., Vierke, G.: Holonic multi-agent systems. Technical Report DFKI-RR-99-03, Deutsches Forschungszentrum für Künztliche Inteligenz - GmbH, Postfach 20 80, 67608 Kaiserslautern, FRG (1999)
9. Hammond, T., Davis, R.: LADDER, a sketching language for user interface developers. Computers & Graphics 29(4), 518–532 (2005)
10. Lieberman, H.: Letizia: An agent that assists web browsing. In: 14th Int. Joint Conference on Artificial Intelligence, pp. 924–929. Morgan–Kaufmann (1995)
11. Locoro, A., Grignani, D., Mascardi, V.: MANENT: An Infrastructure for Integrating, Structuring and Searching Digital Libraries. In: Biba, M., Xhafa, F. (eds.) Learning Structure and Schemas from Documents. SCI, vol. 375, pp. 315–341. Springer, Heidelberg (2011)
12. Lowe, D.G.: Object recognition from local scale-invariant features. In: International Conference on Computer Vision, pp. 1150–1157. IEEE Computer Society (1999)
13. Papaleo, L., Quercini, G., Mascardi, V., Ancona, M., Traverso, A., Lumley, H.D.: Agents and ontologies for understanding and preserving the rock art of mount bego. In: Filipe, J., Fred, A.L.N. (eds.) ICAART 2011. Agents, vol. 2, pp. 288–295. SciTePress (2011)
14. Viola, P.A., Jones, M.J.: Rapid object detection using a boosted cascade of simple features. In: CVPR 2001, pp. 511–518. IEEE Computer Society (2001)
15. W3C. OWL 2 Web Ontology Language Document Overview – W3C Recommendation, October 27 (2009)
16. Zambanini, S., Kampel, M.: Automatic coin classification by image matching. In: Niccolucci, F., Dellepiane, M., Serna, S.P., Rushmeier, H.E., Gool, L.J.V. (eds.) VAST 2011, pp. 65–72. Eurographics Association (2011)

A Proposed Architecture for a Fault Tolerant Multi Agents System Using Extern Agents

Mounira Bouzahzah[1] and Ramdane Maamri[2]

[1] Lire Laboratory, Science and Technology Institut., University Center of Mila, Mila, Algeria
[2] Lire Laboratory, Computer Science Departement, Mentouri University, Constatine, Algeria
m.bouzahzah@centre-univ-mila.dz,
rmaamri@umc.edu.dz

Abstract. This paper proposes an approach that makes it possible to have a robust multi agents system. Generally, a multi agents system has a mission called the global goal that it has to achieve. This global goal can be decomposed into different sub goals. We define two types of goals: the simple sub goal realized by only one agent and the complex sub goal achieved by more than one agent. The agents that realize simple goals formed one class, and they are replicated to assure fault tolerance. Each group of agents in cooperation formed one group called the complex goal class; it uses the exception handling technique to treat the errors. We use extern agents to manage the classes formed and to control the whole system.

Keywords: Extern agents, Petri Nets modelling, Multi agents system's mission, simple sub goals class, complex sub goals class.

1 Introduction

There are two main mechanisms that are used in field of fault tolerance: the replication mechanism which is based, generally, on redundancy, a definition to a replicated software component is given in [01]. Using the replication technique within agents that realize simple goals gives the system the opportunity to satisfy its mission even if some agents are in failure. A failed agent is an agent that cannot execute actions and also cannot send or receive messages. The exception handling mechanism allows the system to treat the errors when they are signalled. So, this technique is more adequate to be applied within complex goals classes.

This paper is organized as follows: the second section describes the general architecture of the proposed system. Section 3 illustrates the different types of the extern agents and describes their activities using Petri Nets mechanism. Section 4 gives some related works. Finally, section 5 concludes our paper and gives an insight concerning our future work.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 282–289, 2012.

2 The System's Architecture

Combining replication and exception handling is proposed by [02] but this work tries to applies the SaGE proposed in [03] on a replication platform DIMAX [04]. Our work tries to use these two mechanisms separately but within the same multi agent system to assure fault tolerance. Each multi agents system has a global mission that it has to achieve; this mission can be decomposed into sub goals. We defined two types of goals:

- A simple goal is realized by one single agent; replication is used within this group because agents are considered as important.
- A complex goal which needs more than one agent to be achieved. The existing of two or more agents in cooperation to realize one complex goal gives us the idea of using exception handling within this type of classes.

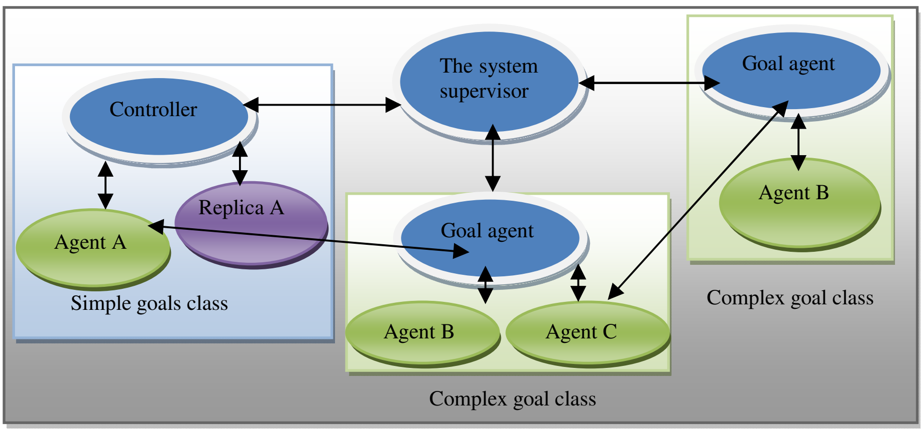


Fig. 1. The proposed system's architecture

Figure 1 gives the general architecture of the multi-agent system, in order to guarantee fault tolerance; we propose to add extern agents.

2.1 The Agents' Classes

Agents' groups or classes are used in different approaches such as: [05] and [06] to facilitate agents' classification. The framework DARX also deals with the notion of groups it uses the replicas groups. This framework associates a group within each replicated agent. Concerning our work, we will create a class that includes agents realizing simple goals and we associate a class for each complex goal.

2.2 Simple Sub Goals Class

The agents that belong to the simple sub goals class are considered as critical agents because if one of them fails the simple sub goal associated to him will not be achieved and the multi agents system can fail in achieving its mission; thus, they are replicated. There are two strategies of replication: active and passive. The active replication is defined as the existence of several replicas which process concurrently all input messages [07]; this strategy provides short recovery but leads to a high over-head. And the passive replication which indicates that only one activated replica processes all input messages and transmits periodically its current state to the other replicas in order to maintain coherence of the replicated group, and to constitute a recovery point in case of failure. This strategy requires less CPU resources. Trough this work, we choose to apply the passive strategy of replication because it is more adequate to be implemented within the JADE platform. We propose in our work that each agent will have only one replica and when the agent fails other copies will be added to the system by an extern agent called the controller. The controller manages the simple goal class and creates replicas following the system's needs.

2.3 Complex Sub Goals Classes

A class is created within each sub goal considered as complex and a goal agent is added to each class in order to treat exceptions or faults. The goal agent is a generic agent that controls the executions, detects failures and assures system recovery. It has a decision function that can ignore an exception signal or consider it depending on the criticality of the action that gives the signal. To handle an exception the goal agent creates for each action existing in the plan associated to the complex goal a list of agents that can execute this action. In the case of an exception signalled and considered by the decision function, the goal agent treats it as follows:

- If the action failed has other agents that can execute it, the goal agent gives request to all the agents that are in the set of action to satisfy it.
- If this action cannot be executed by any agent in the system, then the goal agents uses the programs given by the designer. The designer must initiate the goal agent with some programs that treat known exceptions.
- If the error is too difficult, the goal agent must give an extern signal.

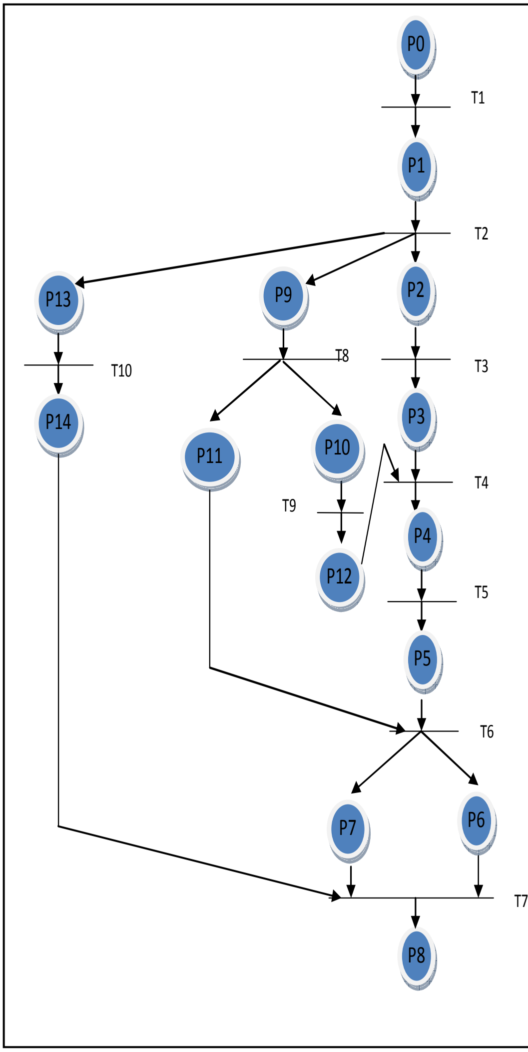
3 The Extern Agents

The management and the control of the system's classes are done by added agents called the extern agents.

3.1 The Controller

It allows agent replication using the passive strategy. This agent verifies if an agent is still alive or it is in case of failure using the idea synchronous environment [08]. It sends control messages to the agents using the time criteria [09]. Thus, a timer called failure _ timer that represents the max response time of the agent, and which will be used to calculate the time. The controller uses the procedure Detection_failure which calculates the response time of agents, if it never exceeds the max response time, a failure is detected. Since the detection of failure, the passive replica will be active and another passive replica is added to the system

Petri Nets [10] are promising tool for describing and modelling system activities. This formalism allows parallel processes modelling. Figure 2 shows the controller model.



- P0- message.
- T1- Receiving the message.
- P1- Message received.
- T2- Checking the type of the message:
- P2- Message asking for contract.
- T3- Establishing contract.
- P3- Contract established.
- T4- Replicating the concerned agent.
- P4- Agent replicated.
- T5- Creating the checking message for the replicated agent.
- P5- Controlling message.
- T6- Initializing the failure time.
- P6- The timer initialized.
- P7-message created
- T7- Sending the controlling message.
- P8- message sent
- P9- Agent's response.
- T8- Checking if the response is at time
- P10- failed agent.
- P11- Agent alive.
- P12- Active replica.
- T9- Activating the passive replica.
- P13- Checking message sent by the agent supervisor.
- T10- Creating a response.
- P14- Response created.

Fig. 2. The controller model

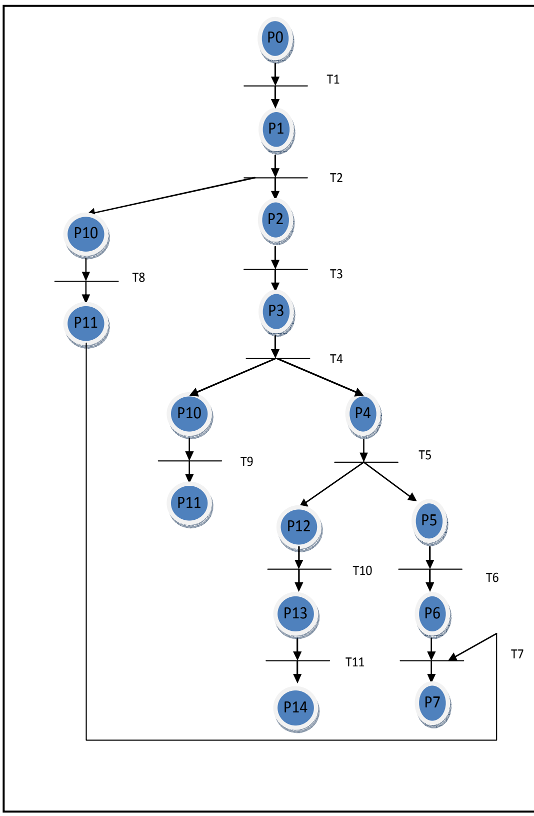
When a message P_0 arrives, the controller checks its nature that can be:

- A contract message P_2 that means a new agent will be added to the simple sub goal class: a contract is established and the new agent will be replicated P_4, the controlling message concerning this agent is created P_5.

- A control message sent by the system supervisor P_{13}: in this case a response is created P_{14} and sent to the super visor.
- An agent response P_9 which confirms that it is still alive or a fault signal if the response time is finished. If the controller detects a failure, it must activate the passive replica $T_9$and add a new replica to the system.

3.2 The Goal Agent

The goal agents are extern agents added to the system in order to treat exceptions that occur in complex goal groups. If an agent is considered as a failed one, the action that it has to achieve gives an exception signal. The following figure describes the goal agent activities using Petri.



- P0- message.
- T1- Receiving message.
- P1- Message received.
- T2- Checking the MESSAGE type
- P2- Failure message.
- T3- Calculating the response
- P3- The result.
- T4- Analyzing the result.
- P4 failure considered.
- T5- Treating the failure.
- P5- Action done by other agents.
- T6- Creating message
- P6- Message created
- T7- sending message.
- P7- message sent
- P8- a checking message sent by the system supervisor.
- T8- creating the response.
- P9- the response created.
- P10-failure ignored.
- T9- ignoring the exception.
- P11- message to ignore
- P12- Action done by one agent.
- T10- handling exception using program's modules.
- P13- Exception handled.
- T11- Recovering the failure.
- P14- exception treated.
- T11- Adding the agent to the complex goal group.

Fig. 3. The goal agent model

The goal agent receives a message P_0 that can be:

- An exception signal which activates the decision function T_3.If the exception is considered then the goal agents must use the techniques described in the previous section to treat it.
- A control message sent by the system supervisor P_8: in this case a response is created T_8 and sent to the super visor.

3.3 The Decision Function

The goal agent that is responsible of errors solving uses a decision function that takes decisions about considering an error or ignoring it. At the designing time each action present in the plan that represent the complex goal must has a propriety that reflect the impact that it has on the realization of the whole complex goal. The designer gives the value of this propriety using different criteria:

- According to the complex goal and the global one, the designer can determine semantic information to define the importance of an action.
- The numbers of agents that executes actions related to the results given by the concerned action has, also, an impact on the criticality of the action.

The propriety that the designer has to associate to each action defines a measure indicates the impact of the loose of this action on the entire goal and it is called criticality [11]. The designer determines action's criticality as a set of ordered and limited values. The min criticality: generally represents the value 0 which is associated within an action that has no impact on the goal to be achieved. However, the max criticality represents a critical action that may cause the failure of the whole goal. The decision function uses the Procedure Division in order to determine the critical actions.

```
Procedure Division ( )
{
Sum_criticalities = 0;
For all action in the goal plan {
  Read C (A); / * C (A) the action A criticality * /
  Sum_criticalities = Sum_criticalities + C (A);
     }
For each action A {
  If (C (A)> = ((Sum_criticalities) / number ofgoal's
actions)
        exception = 1;
  Else
        exception = 0 ;}
/ * Exception = 1 represents a critical action, however,
exception=0 indicates an action with less impact* /
```

The decision function can ignore an action failure when the other actions continue execution, and this action is considered as being less impact. Otherwise, the fault must be treated by the goal agent.

3.4 The System's Supervisor

The Supervisor is also an extern agent that plays a very important role in this approach; it controls the other added agents and replace them by their replicas in case of failure. An agent is considered as failed if does not response at a checking request. In reality, the supervisor controls the added agents and its failure has no impact on the whole system. We propose that the Supervisor has a procedure called history which saves the state of the environment at limited time's intervals. In case of the supervisor failure this information will constitute a checkpoint for recovery.

4 Existing Work

The proposed approaches that use replication are classified into two broad families of fault tolerance [12]. A. Fedoruk and R. Deters [13] introduce a new approach based on transparent replication agents using proxies. A proxy managed communications between a group of agents and other groups of agents and controls the interaction between this group of agents and the environment. A. Almeida et al [14] introduce a methodology for replication using the framework DARX [15]. This approach treats the case of cooperative systems and it obliges agents to interrupt their executions in order to evaluate their criticality. The use of replication gives the system the opportunity to achieve its global goal even in case of fault; but replicating all the agents of the system can cause difficulties when controlling the system.

The mechanism of exception handling is proposed through different works and using different strategies. [06, 16] propose specific systems for exception handling. The approach described in [06] introduces the concept of the supervisor which has the role of a handler for a group of system's agents. Hagg [16] proposes a strategy for exception handling using sentinels. Sentinels are guardian agents which protect the multi-agent system from failing in undesirable states. They have the authority to monitor communications in order to react to fault. This approach is costly in terms of computation and communication and it causes point of failure since sentinels, also, are subject of fault. The system described in [03] gives more important to agents' communications in terms in request/ response. An exception is treated by handlers that try to solve the problem locally, at this function has information concerning the criticality of each tasks executed and it judges an exception if it is ignored or token in consideration.

5 Conclusion

This paper describes the proposed architecture of a fault tolerance multi agents system. This work is characterized by the use of two different technique applied in fault tolerance field which are: replication and exception handling. It is the first time that they are used separately but within the same system. Concerning our future works, we will try to use learning algorithms to allow the goal agents to learn from their experiences and to facilitate exception handling. The implementation will be done using the JADE platform.

References

1. Guessoum, Z., Briot, J.P., Faci, N., Marin, O.: An adaptive replication mechanism for SMA fault tolerant. JFSMA (2004)
2. Kchir, S.: Gestion des Exceptions dans un Système Multi Agents avec Réplication. Master 2, Laboratoire d'Informatique de Robotique et de Micro-électronique, Montpellier (2010)
3. Souchon, F., Christophe, D., Christelle, U., Sylvain, V., Jacques, F.: SaGE: une proposition pour la gestion des exceptions dans les system multi agents. Internal repport-LIRMM-02205 (2002)
4. Faci, N., Guessoum, Z., Sailor, O.: DIMAX: A Fault Tolerant Multi- Agent Platform. In: SELMA 2006 (2006)
5. Kumar, S., Cohen, P.R., Levesque, H.J.: The adaptive agent architecture: achieving fault-tolerance using persistent broker teams. In: The Fourth International Conference on Multi-Agent Systems (ICMAS 2000), Boston, MA, USA,(2000)
6. Anand, T., Robert, M.: Exception Handling in Agent Oriented Systems. Springer-Verlag New York, Inc., New York (2001)
7. Wiesmann, M., Pedone, F., Schiper, A., Kemme, B., Alonso, G.: Database replication techniques: a three parameter classification. In: Proceedings of 19th IEEE Symposium on Reliable Distributed Systems (SRDS 2000), Nüenberg, Germany (2000)
8. Fischer, M., Lynch, N., Patterson, M.: Impossibility of distributed consensus with one faulty process. JACM (1985)
9. Huang, Y.F., Chuang, M.H.: Fault tolerance for home agents in mobile IP. Computer Networks 50 (2006)
10. Bourdeaud'Huy, T.: Techniques d'abstraction pour l'analyse et la synthèse de réseaux de petri. Doctorat theses, Lille University (2004)
11. Almeida, A., Aknine, S.: A Predictive Method for Providing Fault Tolerance in Multi-Agent Systems. In: Proceedings of the IEEE/ WIC/ACM International Conference of Intelligent Agent Technologie (IAT 2006), France (2006)
12. Baroudi, R.: An Approach for Estimating Criticality Agents Based on Fuzzy Classification. Master theses, Computer ISI, LIP6, Paris IX University (2008)
13. Fedoruk, A., Deters, R.: Improving fault-tolerance by replicating agents. In: Proceedings AAMAS 2002, Bologna, Italy, pp. 144–148 (2002)
14. Almeida, A., Aknine, S.: Method of replication based on plans for the fault tolerance of multi-agent systems. JFSMA (2005)
15. Marin, O., Marin, B., Sens, P.: DARX: A Framework for Tolerant Support of Agent Software. In: Proceedings of the 14th International Symposium on Software Engineering Reability. IEEE (2003)
16. Hagg, S.: A Sentinel Approach to Fault Handling in Multi-Agent Systems. In: Proceedings of the Second Australian Workshop on Distributed AI, Cairns, Australia (1996)

GeoHash and UUID Identifier for Multi-Agent Systems

Zoran Balkić[1], Damir Šoštarić[2], and Goran Horvat[2]

[1] Department of Automation and Process Computing
[2] Department of Communications
Faculty of Electrical Engineering, J.J. Strossmayer University of Osijek
Kneza Trpimira 2b, 31000 Osijek, Croatia
{zoran.balkic,damir.sostaric,goran.horvat}@etfos.hr

Abstract. This paper documents the case to address the distributed data storage and widely used unique identifiers which are in use from the beginning of databases and backend systems. The importance of uniqueness for object system identifiers is crucial for efficient data retrieval, storage and comparison. We are representing the new method for objects tagging using synergy mechanism between well known Geohash algorithm and Universally Unique Identifiers we call GHUUID (Geohash+UUID) gaining the agent system spatio-temporal identification of all data available. Multi-Agent systems in their nature strongly rely upon distributed databases across vast amount of data. Some of the interoperability issues are solved with usage of semantic web principles which can be cumbersome to implement and maintain. Most systems need the basic space-time correlation which can be the corner stone for the efficient analysis and data aligning regardless of objects stored and their nature.

Keywords: Multi-Agent mechanism design, distributed unique identifiers, geo location awareness, data filtering and analysis.

1 Introduction

Since the beginning of the information technology era there has been a necessity to implement unique identifiers over stored data. Using identifiers has become the base for successful database design as well as different programmed solutions in all programing languages and data formats. Without them it would be practically impossible to address the specific object in the local data space including, relational, document oriented or hierarchical databases in use today. Identifiers are the basic structure in every modern database implementation and the core of every relational database ensuring the consistence of data and relational paradigm using primary and foreign keys for schemas and object reference. Multi-Agent systems are the one that benefit from this methodology the most since identifiers are essential for successful data retrieval over different database tables as well as data update and deletion. Graphical user interfaces of all kinds (web, fat clients, embedded systems etc.) strongly rely on identifiers uniqueness and consistency. All modern programming languages and data formats use them heavily during creation and annotation process for automatic programming models generation. The most important characteristic of

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 290–298, 2012.

the ID's is their uniqueness and the fact that once identifier is generated can not be duplicated at any time, regardless of object deletion or move in the data instances space. All database solutions today implement this mechanism to ensure that ID's, often called the "key", should not be duplicated which would lead to data corruption and inconsistency. Database relations have ID's in their core allowing classic relational database to operate.

Usually the ID's of the objects in the database table are used for this purpose, not offering any global identifier in the global information space. With the introduction of UUID's this goal is implemented and conducted in the way that globally unique identifier is satisfied, offering the true uniqueness across different systems without need for centralized authority for issuing those identifiers.

UUID is time-based generated identification mechanism and, as such, highly precise and collision free.

With the rise of Geo location aware systems lately, the importance of change in agent's infrastructure design is evident.

2 Motivation

Multi-Agent system (MAS) is the system that consists of a different number of standalone agents which should interact with each other. The main requirement for their operation is cooperation, coordination and negotiation like people do in their everyday life. As people are mainly interested in space and time information of the object, this information is the main goal for implementation into Multi-Agent systems on the global identifier level.

One among the first MAS applications was distributed vehicle monitoring (DVMT) [1,2] where a set of geographically distributed agents monitor vehicles that pass through their respective areas, attempt to come up with interpretations of what vehicles are passing through the global area, and track vehicle movements. The DVMT has been used as an MAS test bed.

To enable greater decision making process simpler we propose Geohash+UUID (GHUUID) mechanism which would take location based data, obtained from the network or explicitly from GPS data, with UUID functions. Both GeoHash and UUID functions are implemented in most modern database systems and widely available as standalone functions in modern programing languages for both, client and server side usage.

Searching the Internet and distributed databases for a specific query can be a long and tedious process thus assembling the right tool for searching and retrieval of data between higher correlated data based on GHUUID is of high importance.

Internet or network enabled agents, typically needs to gather information that would require synthesizing pieces of information from various available information sources and GHUUID is a perfect tool for simplifying that task.

According to [3] Multi-Agent based approach to clustering, that harnesses the processing power of a collection of "clustering" agents, to produce a "best" set of clusters given a particular clustering problem does not exist. General purpose, best clustering algorithm that is suited for all data is still unavailable, therefore GHUUID is one of the best methods providing instant alignment tool that could enhance Multi-Agent data mining environments. The accuracy of a set of Multi-Agent clusters requires pre-labeled data which needs to be manually added to the system that entails an undesirable overhead. In the [3] two measures are introduced for Multi-Agent systems clustering, Within Group Average Distance (WGAD) and total Between Group Distance (BGD) which needs data mining ontologies for efficient Multi-Agent Data mining (MADM).

The main contributions of this paper are as follows:

- Introduction of new identifier that consists of Geohash encoding and UUID unique identifiers which eliminates the usage of any external labeling of data or data mining ontologies.
- A generic approach to Multi-Agent based clustering techniques to identify a best set of clusters using a GHUUID which relates agents according to space and time measurements
- Development of the test environment based on Java Enterprise Edition (JEE) with Spatial enabled MySQL database server for GHUUID showcase

3 Geohash

Geohash is a latitude/longitude geocode system invented by Gustavo Niemeyer for encoding/decoding (lat,lon) pairs in a compact form.

The Geohash algorithm [4,5] can be used to divide geographic regions into a hierarchical structure. A Geohash is derived by interleaving bits obtained from latitude and longitude pairs and then converting the bits to a string using a base-32 character map. A Geohash string represents a fixed spatial bounding box. For example, the latitude and longitude coordinates of *45.557, 18.675* falls within the Geohash bounding box of "*u2j70vx29gfu*". Appending characters to the string would make it refer to more precise geographical subsets of the original string.

Encoding the position takes only (lat, lon) pairs neglecting the third coordinate (altitude) and as such it is unaware of real 3D position of the object in space.

As a consequence of the gradual precision degradation, nearby places will often (not in some edge cases, equator or a meridian) present similar prefixes. The longer a shared prefix is, the closer the two places are.

To obtain the latitude and longitude bits from an initial pair of coordinates representing a target point in space, the algorithm is applied recursively across successively more precise geographical regions bounding the coordinates. The remaining geographical area is reduced by selecting a halfway pivot point that alternates between longitude and latitude at each step. If the target coordinate value is greater than the pivot, a 1 bit is appended to the overall set of bits; otherwise, a 0 bit is appended. The remaining geographic area that contains the original point is then used in the next iteration of the algorithm. Successive iterations increase the accuracy of the final Geohash string. An appealing property of the Geohash algorithm is that nearby points will generally share similar Geohash strings. The longer the sequence of matching bits is, the closer two points are.

This property is exploited in GHUUID to support simple range-based spatial queries that return data in a given Geohash region, allowing agents to specify more or less precise hashes to select smaller or larger areas. It is also possible to use Geohashes for quick proximity searches. In addition to queries, Geohashes can also be used to group similar data. If a collection of data gets too large, simply using more precise Geohashes allows the system to create more specific, and therefore smaller, groupings of data. This property also allows quick retrieval of similar data blocks for use in computations.

In this paper database GeoHash encoding function is created for the MySQL database since it is not part of the database distribution. Standard database stored procedure, in this case, is deployed into the database which is triggered upon any data creation or update and stored in the separate table for faster data retrieval, search and comparison.

4 Universal Unique Identifier (UUID)

Universal Unique Identifier (UUID), also known as GUID (Globally Unique IDentifier) is generated according to [6]. A UUID is designed as a number that is globally unique in space and time. Two calls to UUID() function are expected to generate two different values, even if these calls are performed on two separate computers that are not connected to each other.

Stated in [7] UUID is a 128-bit number represented as an utf-8 string of five hexadecimal numbers in aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee format and requires no central registration process:

- The first three numbers are generated from a system timestamp.
- The fourth number preserves temporal uniqueness in case the timestamp value loses monotonicity (for example, due to daylight saving time).
- The fifth number is an IEEE 802 node number that provides spatial uniqueness. A random number is substituted if the latter is not available (for example, because the host computer has no Ethernet card, or we do not know how to find the hardware address of an interface on your operating system). In this case, spatial uniqueness cannot be guaranteed. Nevertheless, a collision should have very low probability.

Several variants and versions of UUID generated strings exist today:

- Version 1 - MAC address based with the time at 100 nanoseconds intervals since the adoption of Gregorian Calendar
- Version 2 - DCE security with the upper byte of the clock sequence replaced by the identifier for a "local domain"
- Version 3 - MD5 hash version uses scheme deriving a UUID via MD5 from URL
- Version 4 - relies on the generation of random numbers
- Version 5 - SHA-1 hashing instead of MD5 like Version 3

Currently, the MAC address of an interface is taken into account only on Linux Operating systems. On other operating systems, most systems use a randomly generated 48-bit number.

Since we are interested only in "time" part of the UUID strings for the spatio temporal analysis we are using the Version 1 for the construction of GHUUID identifiers.

Standard Java library does not contain UUID generator which can generate timestamp dependable UUID strings, only Versions 3 and 4 due to lack of means to access MAC addresses using pure Java before version 6.

Even though UUID string could be generated server side using third party Java libraries, in the test process we have used built in UUID function of the MySQL database server which eliminates the external generation of UUID strings thus speeding up the system. Extraction of the "time of creation" from the encoded UUID string is implemented in Java as standalone utility class which is used in the developed JEE application.

5 Geohash UUID (GHUUID)

In this paper we propose the new system for generating system wide global identifiers with all characteristics essential for the high quality identifier that is spatio-temporal aware. GHUUID consists of two parts, first one id GeoHashed longitude/latitude pair of data which can be obtained from either agent, server side system or directly from databases and the other, standard UUID that could be generated on the backend side of the Multi-Agent system. In the case UUID is generated on the backend side, it contains the IEEE 802 node number that provides additional spatial data, even though this information could be of less importance to the agent itself.

GHUUID's in essence are variable length data depending on Geohash precision encoding. In most cases the significant precision of uniqueness is 192-bits (Geohash – 64-bits and UUID – 128-bits) long.

Example of GHUUID looks like:
(u2j70vx29gfu-2d004620-3fc7-11e1-b86c-0800200c9a66)

- First part: *u2j70vx29gfu* represent GeoHashed lat/lon information that corresponds to the initial geo coordinates of *45.557, 18.675*.
- The second part: *2d004620-3fc7-11e1-b86c-0800200c9a6*6 is generated UUID that contains a timestamp taken at Sunday, January 15, 2012 10:20:48 PM GMT.

Multi-Agent systems operating in the way of using GHUUID could simply from those identifiers extract information needed for spatio-temporal awareness over position in the real world and the timestamp for the data either creation or replacement.

GHUUID data can be directly shown in the client Geo visualization software through standard mapping graphical user interfaces with direct links to the object data that are stored on the agents or central system of communication.

Further on, as GHUUID contains timestamp data it could also be presented in the timemap clients giving us the spatio-temporal presentation straight from the system identifiers (Fig. 1.) regardless of the nature, context or data information of the object visualized.

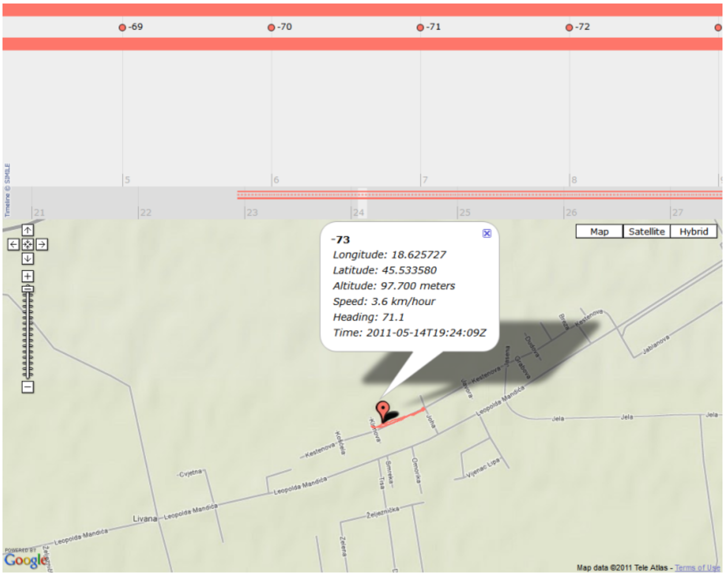


Fig. 1. Timemap view of the system GHUUID identifiers for mobile Agent with corresponding trajectory

5.1 GHUUID Prototype Testing Application

Prototype application is developed using Java Enterprise Edition using Servlets and Java Server pages, deployed into standard Tomcat Java server coupled with MySQL database server containing spatial extensions.

We have chosen Java for the compatibility with the JADE [8] a software framework that can be considered one of the most known and used software frameworks for the MAS development and possible future integration with it.

Full system test was made on the data set that consists of 6988 mobile Multi-Agents and 52 static Multi-Agents which are static in space thus changing only their time component. All mobile Agents have their data collected in the time period of 2 minutes with the pseudo random pattern across the space.

GeoHash is calculated based upon current location of the Agent (Fig 2.) with coordinates (*45.534, 18.628*) and passed to the application in full GeoHash encoded form (*u2j70vx29gfu*) constrained by the number of characters used to match the neighboring Agents. This process is iterative in the manner of shortening the Geohash from six to one character (*u2j70v, u2j70, u2j7, u2j, u2, u*) giving greater number of matching Agents in space and speeding up the query execution.

GHUUID is consisted of two identifiers and their relation in the process of determining the closest Agents in space and time can vary from 0-50% in both directions from space to time and vice versa as shown in (Fig 3).

Calculated GHUUID weight ratio (%) visualized in (Fig. 4.)

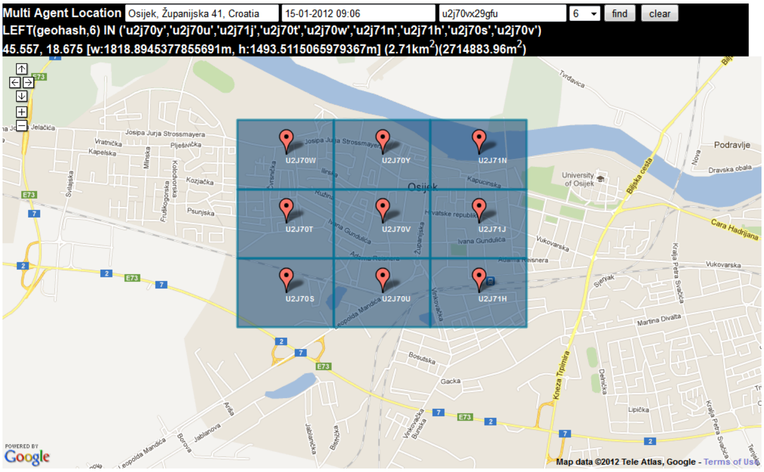


Fig. 2. Map view of the searched GHUUID identifiers

- 20% in the favor of Location
- 40% in the favor of Location
- 50% in the favor of Location or Time (used in this test case)
- 40% in the favor of Time
- 20% in the favor of Time

Multi-agents that are more related in space and time are colored red and ones that are less related are colored green (Fig. 3.).

Regardless of the type and nature of stored data per Agent efficient relation is calculated in real time and metrics of queries sent to the database system is shown in Table 1. The results of the metrics brings interesting conclusion that searching the wider area, which gives more matching results, takes less time than more specific and narrowed queries which is not the case in the classic Geo spatial queries which usually take more time with the larger data sets.

Table 1. Metrics of the system performance in Agent number per GeoHash lenght (characters) and the query results in seconds

GeoHash / Agents	1	2	3	4	5	6
1	0.0101	0.0104	0.0106	0.0386	1.3634	1.3736
2	0.0094	0.0097	0.0102	0.0394	1.3705	1.3722
3	0.0088	0.0094	0.0102	0.0373	1.3621	1.3664
4	0.0086	0.0091	0.0101	0.0373	1.3568	1.3697
5	0.0083	0.0089	0.0102	0.0371	1.3555	1.3705
10	0.008	0.0092	0.0100	0.037	1.3501	1.3711
100	0.0098	0.0103	0.0110	0.0409	1.4626	1.4705
1000	0.0144	0.0151	0.0161	0.0599	2.1418	2.1533

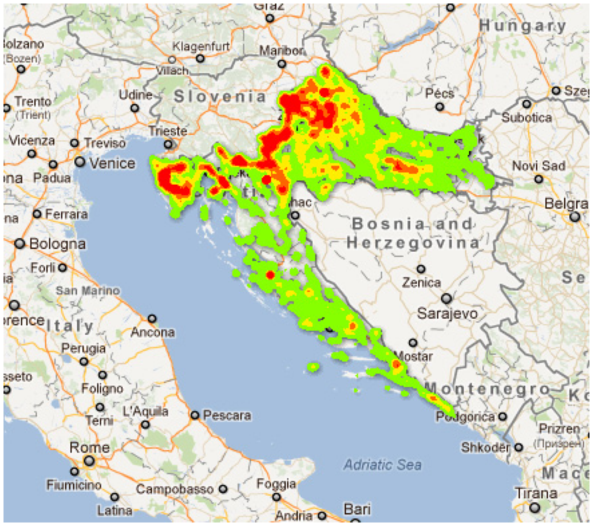


Fig. 3. GHUUID relation heat map

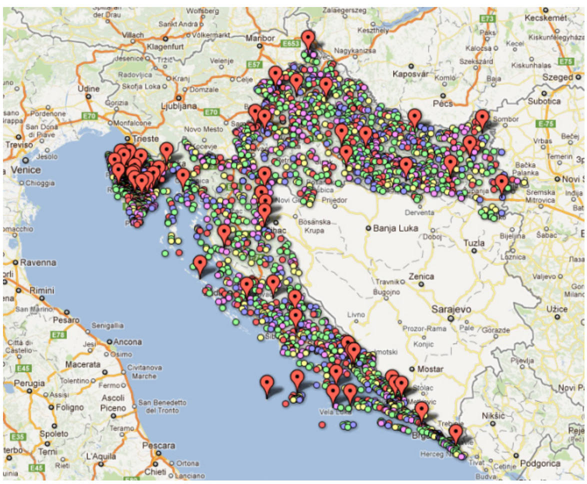


Fig. 4. GHUUID ratio visualization

6 Conclusion

Design and implementation of agent systems is difficult task. All the standard problems that exist with building traditional distributed, concurrent systems are emphasized with additional requirements like smart interactions between autonomous components. The effectiveness of the implemented MAS is therefore more important than before.

Implementing GHUUD generation, sorting and comparison methods into Multi-Agent systems leads to simplification of the Urban Search and Rescue (USAR) domains and Geospatial Multi-Agent Systems (GMAS) which rely upon contributing agents for Multi-Agent Path Planning (MAPP) [9].

Standard algorithms such as A* algorithm, E-GAP, Potential Field method (PFM) and CPAD algorithm as stated in [10] could be simplified with the introduction of GHUUID identifier.

The GHUUID method is not limited only to Geo tagged objects, static or dynamic ones; it can be used in the systems without particular Geo referential system with conversion of referential spatial systems, such as two dimensional Cartesian coordinate system, into the Geo datum enabled WGS84 or country specific geo datum.

Using our truly global identifier for data/object Meta tagging can simplify the MAS structure, give the industrial-strength to agent interaction, enhance the visualization development and predict the other agent's actions under different circumstances.

GHUUID enabled agents could perform autonomous data identifiers generation, filtering and enable smarter data flow over the Internet and private networks, obtaining only the data that are closer thus more connected.

References

1. Durfee, E.: Planning in Distributed Artificial Intelligence. In: O'Hare, G.M.P., Jennings, N.R. (eds.) Foundations of Distributed Artificial Intelligence, pp. 231–246. Wiley, New York (1996)
2. Durfee, E.H., Lesser, V.: Negotiating Task Decomposition and Alloca-tion Using Partial Global Planning. In: Gasser, L., Huhns, M. (eds.) Distributed Artificial Intelligence, vol. 2, pp. 229–244. Morgan Kaufmann, San Francisco (1989)
3. Chaimontree, S., Atkinson, K., Coenen, F.: Clustering in a Multi-Agent Data Mining Environment (2010)
4. W. Contributors, Geohash, Wikipedia.org (2011)
5. http://geohash.org/site/tips.html
6. A Universally Unique IDentifier (UUID) URN Namespace, http://www.ietf.org/rfc/rfc4122.txt
7. http://dev.mysql.com/doc/refman/5.0/en/miscellaneous-functions.html#function_uuid-MySQLUUIDsupportfunction
8. Bellifemine, F., Caire, G., Poggi, A., Rimassa, G.: JADE: a Software Framework for Developing Multi-Agent Applications. Lessons Learned. Information and Software Technology Journal 50, 10–21 (2008)
9. Gaber, H., Amin, S., Salem, A.-B.M.: Geospatial Multi-agent System for Urban Search and Rescue (2011)
10. Gaber, H., Amin, S., Salem, A.-B.M.: A Combined Coordination Technique for Multi-Agent Path Planning. In: Proc of ISDA (2010)

Multi-agent Power Management System for ZigBee Based Portable Embedded ECG Wireless Monitoring Device with LabView Application

Damir Šoštarić[1], Goran Horvat[1], and Željko Hocenski[2]

[1] Department of Communications
[2] Department of Automation and Process Computing
Faculty of Electrical Engineering, J.J. Strossmayer University of Osijek
Kneza Trpimira 2b, 31000 Osijek, Croatia
{damir.sostaric,goran.horvat,zeljko.hocenski}@etfos.hr

Abstract. The techniques of multi-agent system bring intelligence and flexibility to embedded agent/multi-agent embedded system connected to internet presents a great advantage. Advantage of such multi-agent systems improves the use of expanded infrastructure. Installing simplified agents in embedded systems was shown necessary due to increasing use of embedded devices. Multi-agent power management system is based on battery control and anticipation of replacement. Wireless transmission of ECG (electrocardiogram) signal via ZigBee (XBee modules) brings some problem into focus. This paper presents concept realized and tested on real equipment. Using smart mobile phone (today a widely used device) interaction/actoric between end user embedded agent and embedded master agent can give feedback about end users health in real-time. Similar, off-line monitoring device exists, but not connected to the network (Holter). Monitoring of patients in real-time can be enabled by such device that exhibits wireless communication and allows transmission of real-time source signal. The agent on which this paper refers to is program/firmware function in program code in small embedded system for monitoring ECG signal. Simplified agent activates the watchdog for battery alert. Main focus is to control the power consumption of WSN (Wireless Sensor Node). Power management has the intelligence middleware and allows timely to respond and inform the end users. Artificial intelligence is integrated in master agent that is element of embedded system cloud and a primal high level layer. Secondary layer is integrated in dedicated servers which respond to device clouds. LabVIEW application for signal processing provides robust and efficient environment for resolving ECG signal processing problem. These tool/application can be also used in other biomedical signal processing applications such as Magnetic Resonance Imaging (MRI) and Electroencephalography (EEG).

Keywords: embedded agent and multi-agent system, power management, ECG device, ECG application, embedded device cloud.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 299–308, 2012.

1 Introduction

Electrocardiography today is still indispensable diagnostic method in the detection of various cardiovascular diseases with adults and children. With this method an electrical activity of the heart during the cardiac cycle is recorded and interpreted. The history of these diagnostic methods dates back to 1887 when psychologist Augustus Désiré Waller used capillary electrometer and got first electrocardiogram. Netherland's doctor and a physiologist Willem Einthoven (1903) constructed a precise galvanometer with wire and specified standard drains and thereby set the standards of routine examination method. The whole electrocardiography is based on simple addition and subtraction of vectors and registering additive vectors on the body surface. Myocardial depolarization in the electrocardiogram corresponding QRS complex showed on Fig. 1. P wave is atrial depolarization, T wave or ST segment represents the repolarization, [1].

Fig. 1. QRS complex

With standard electrocardiogram according to Einthoven triangle, Fig. 2., the movement of the additive vectors is monitored in two planes: frontal (standard leads I, II, III, AVL, AVF, AVR) and sagittal (precordial leads V1-V6), [1]. Besides the standard electrocardiogram recording, today in the diagnosis of various heart diseases other modifications are also used. ECG in other modification seeks to capture paroxysmal events such as transient ischemia or paroxysmal arrhythmias which may endanger the life of the patient. With *Holter* ECG device a 24-48 hour off-line monitoring is enabled.

Telemetry represents two options for medium access. Through the use of *XBee Series 2* and *XBee PRO Series 2* (S2B-programmable *freescale* microcontroller) embedded system is realized. Integration method of embedded agent on end user device has requirements for source-code development and firmware on demand of multi-agent manager system which is then polled in the form of middleware.

Embedded system usually contains a low power microcontroller. *XBee* modules contain the *freescale* programmable microcontroller and connection is proposed. *Body*

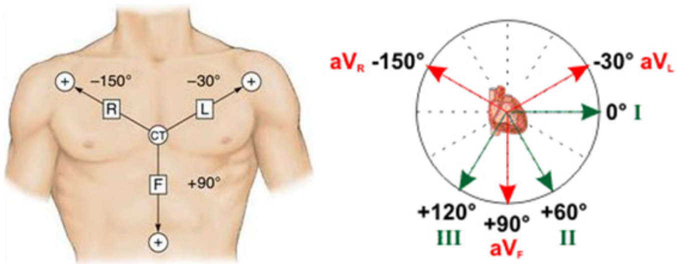


Fig. 2. Vectors in two planes of Einthoven triangle, [1]

network area can have interaction through *ZigBee* (default) and *Bluetooth* (used small module).

In case of a *Holter* device, the used *microSD* card is not suitable for on-line monitoring [2, 3]. *Holter* device can be converted into a wireless monitoring device by installing *XBee* module. Using *microSD* card's Serial Peripheral Interface (SPI) the data can be bridged into a standard serial RS232 interface. The incoming data on the SD card is routed through the wireless channel and sent to one of the device clouds.

Embedded agent located in end user node responds to the monitored value (usually current) of power consumption and informs main dedicated device cloud server. Except the device cloud server on high level of layer, a primer high level layer of embedded device cloud exists. Such cloud system is identical to network oriented hierarchy of technical process in many automated factories. That cloud system has operating system (OS) and can contain artificial intelligence. Algorithms and custom self learning segments become easier for integration through graphical programming interface [4].

Embedded systems are generating increasing volumes and variety of data. These devices are typically found on the "edge" where the "machine" meets the real-world. Many examples of embedded systems "on-the-edge" can be found across a range of industries such as industrial automation, process control, manufacturing, mining, farming, energy, *medical*, consumer electronics, air-traffic control, transportation, warehousing, gaming, and home automation, among others.

Historically, edge-device (end user) networks have been isolated from the rest of the corporate information-technology (IT) infrastructure and the Internet. With the emergence of cloud computing, IT compute and storage resources can be provisioned on-demand, without human intervention. The resource usage can be measured, and the resource pool can be elastically scaled up or down to match the demand. This is resulting in a drastic reduction in the cost of IT, and creating an opportunity for new types of applications that become possible by connecting the edge devices to the cloud core.

The elastic and scalable cloud-computing infrastructure is perfect complement for processing massive amounts of data generated by the edge devices, that can vary based on the real-world demands. Once the edge data from a variety of sources enters the cloud infrastructure, "Big Data" techniques can be used for intelligent processing of massive data volumes. The new types of applications that become possible include monitoring and management of the edge devices, real-time analytics and data mining, ability to match pricing with demand, condition-based predictive maintenance, controlling/influencing real-world behavior, and new value-add services.

2 Implementation of Embedded Agent and Global Multi-Agent System Infrastructure

Integration of an embedded agent in the end user device is performed by compiling the software code. This code contains the algorithm implemented through one or more functions. Recording device for ECG signal has a programmable microcontroller which in itself may recursively read the state of power management. Checking the state of the battery voltage, and power consumption RF part of the microcontroller can achieve a sleep mode that satisfies the working conditions. Alerting an individual agent can also encourage the communication of other agents in the range ZigBee environment. Body PAN area represents the blue area in Fig. 3 with four scenarios of end users:

Nr.1. Represents embedded device to measure the ECG signal and it is the half duplex system in terms of ZigBee infrastructure. Appending computer allows the user to login and supervise measured parameters via on-line SCADA (supervisory control and data acquisition) system. If the user has no access to the computer and the ZigBee network is not available this is considered to be a disadvantage.

Nr.2. Such end users system represents an improvement compared to the previous system because the user is enabled almost real-time communication via SMS message and alarming system with personal physician. Additional integration of GPS modules in end device enables the accurate positioning of the patient if wireless system records hart disorder.

Nr.3. this scenario using the optional USB *Dongle XSticker* monitoring and timely response is secured, although there is no ZigBee network toward coordinator. The above scenario applies to the use of computers in the field range of ZigBee transceiver.

Nr.4. fourth scenario involves two wireless technologies: ZigBee and Bluetooth. When a ZigBee network is unavailable with the Bluetooth module integrated in the end device its allowed communication to the mobile device or computer. With the development of Android applications is enabled communication by 3G and SMS services for data exchange as well as informing the feedback from the embedded device cloud.

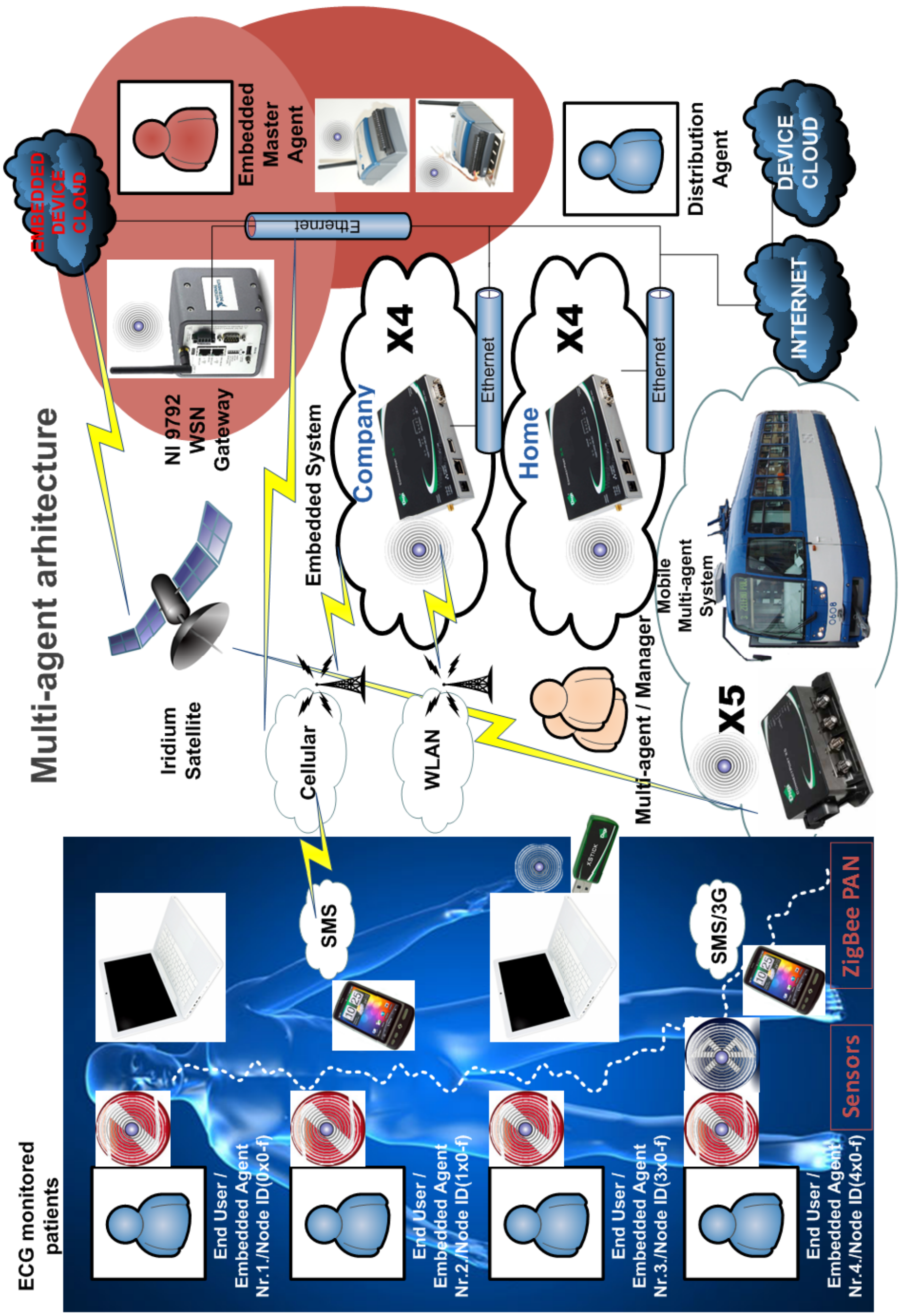
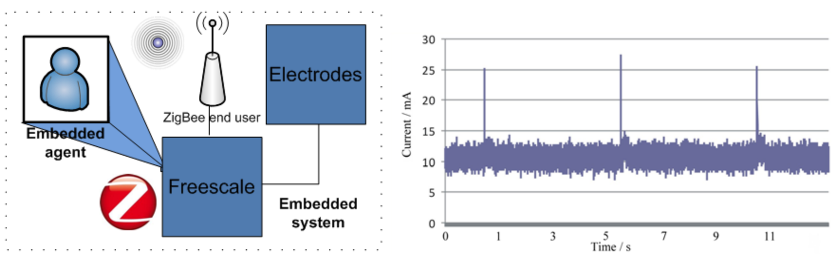


Fig. 3. Global multi-agent system infrastructure

In order for agents to exchange packets within the Cellular and WLAN integrated Multi-agent manager system who "squats" and wakes up to the individual required interrupt routines. On the first upper layer it is possible to realize Company and Home interfaces which are identical to the Smart House embedded systems. Since they are static, they are linked via Ethernet and dual 3G/HSDPA module [5]. Using dual channel information from the end user to device cloud the data is delivered to higher layer of embedded device cloud. Dynamic link for ZigBee is foreseen with installation of a mobile multi-agent system. Mobile agent/multi-agent system is connected via three channels: Cellular network, WLAN and via Iridium satellites. It is also reported as an ConnectPort X4 device cloud, but a mobile agent ConnectPort X5; [5] itself is dedicated embedded router device to cloud.

Multi-agent manager system represents the second communication layer while the Distribution Agent is responsible for monitoring data within the cloud device. The highest hierarchical element is *Embedded Device Cloud* precisely because on the embedded device itself is installed a cloud operating system support. This provides greater distribution and data security with respect to dedicated servers.

Embedded agent on the ECG device is presented in Fig. 4 a). Power consumption of end device and agent reaction is showed on Fig. 4 b), [6]. Microcontroler "wake-up" from sleep mode (agent reaction on external interrupt) is realized in program code in few lines of code. Each peak on Fig. 4 b) represents the data sending. Within the *Freescale* microcontrollers one or more functional routines exists: checking the consumption and calculating and assuming the battery life. Each embedded system is available to the dispatcher (Embedded master agent) who shares priorities to the distribution agent in charge of supervising the activities of - device cloud.



a) b)

Fig. 4. a) Laboratory model of end user device, b) End device power consumption

3 Agent and Multi-agent Power Management System

Commonly monitored parameters are temperature, humidity, pressure, vibration intensity and finally vital body function. With vibration is possible to measure "g" force in all three axis. Additional gyroscope integrated on developed board gives information about drift and detection of fast motion of body. There is a need for constant supervision in medical ECG monitoring applications. With prolonged exposure and use of the device there is additional need for using an agent to alert the consumed energy of batteries. Power management can be integrated on each end user device or on the first upper layer. It is customary that on a first upper layer is located a multi-agent manager in different platform systems. The first available platform definitely must include the ZigBee coordinator, which is located in (CPX4) *ConnectPort* X4 and (CPX5) *ConnectPort* X5. This kind of scenario (Fig. 3) also represents a mobile multi-agent structure. By measuring voltage and with indirect current measurement system is able to assume an active working hours of battery in a particular mode. With application for power management it is possible by prediction establish battery discharge curve.

3.1 Application for Power Management

Application for consumption control is typically integrated into an agent. Agent in end device can be custom programmed via direct programming of wireless sensor node (WSN) or uploading python script remotely, Fig. 5. Python file is programmed in CodeWarrior and control is available for agent, multi-agent and mobile multi-agent systems. Custom programmed uploaded python file represent inputs of voltage and current (indirectly measured from one of A/D inputs). This input values with integrated pole placement regulator, build dynamic logout/actor for transceiver RF hardware unit. Possibility of installing applications with GUI (graphical user interface) and API (application programming interface) is enabled by remote access to virtual embedded server residing on the coordinator (CPX4 or CPX5). Using the web interface it is possible to modify the firmware and achieve custom view. In addition to this layer, applications for power management can be dislocated into a higher layer, which is available to embedded master agent. Such agents are integrated into the operating system and have more processing power because they are not limited by power consumption. For example, the platform uses Crossbow coordinator that is running on Debian Linux, [7]. Terminal access allows installation of additional application, which is the main GUI interface for control of the measured parameters.

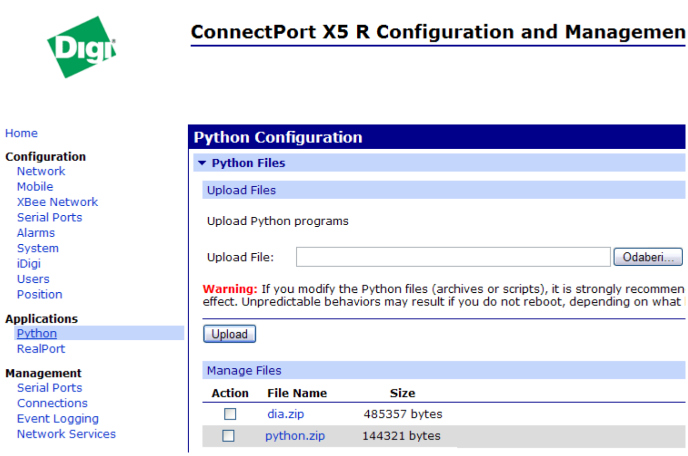


Fig. 5. Remotely web interface for uploading python file

3.2 Location of Embedded Agent in Microcontroller Architecture

The architecture of end user device is based upon a chosen core of the microcontroller. Concept note is based on two electronics application methods, Fig. 6. The first version of the *XBee Pro S2B* module represents a solution to the IOA (Instrumentation Operation Amplifier). *XBee PRO S2B* solution is better because it contains fewer components, and an integrated *Freescale* microcontroller into the module. Series 2 has a drawback as it requires additional microcontroller resulting in additional energy resources. PRO module offers a longer range transmission than Series 2, however the energy consumption is increased accordingly. Additional microcontroller used alongside with Series 2 is an Atmel's AVR architecture, Fig. 6, while the PRO series includes an integrated *Freescale* microcontroller. Architecture and location of embedded agent is shown in Fig. 7, where the agent is located between the processing core and peripheral data input unit. Third agent interface represents a link towards the central power unit and it is defined through various modes, depending on the energy state. Through the compiled software code (ANSI C) the operating system (Firmware) is the main core of the kernel and the real-time system.

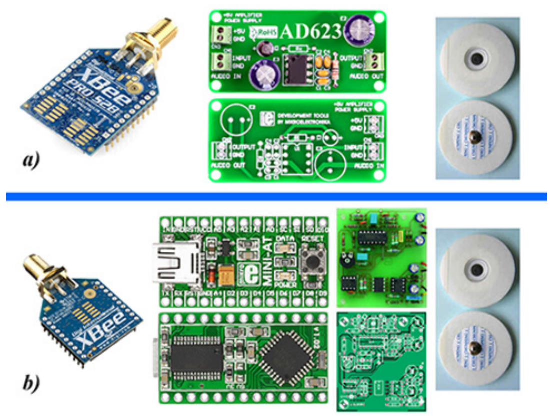


Fig. 6. Hardware scheme of wireless system: XBee PRO S2B, b) XBee Series 2, [8]

The laboratory sensor node system (end user device) can be expanded with ZigBee or Bluetooth module. The central processing unit block represents a microcontroller which includes memory (SRAM, ROM and EEPROM) and the data processing unit. The central power unit provides necessary voltage levels for the microcontroller and connected peripheral devices.

Microcontrollers' clock is programmed to 8 MHz and a corresponding external oscillator is used. The ability of the laboratory sensor node to work in real-time is one of the most important features of the presented system.

When the code with embedded agent function is inserted in the microcontroller, certain memory locations become active. The microcontroller disposes with RAM and ROM memory accordingly. Possibly of additional control is seen through the use of watchdog timer an dislocation of the embedded agent into the EEPROM memory segment. The possibility of additional control is provided through activation of the watchdog timer and dislocation of embedded agent in the EEPROM memory, which does the forthcoming dynamics and communication with the ROM / FLASH memory.

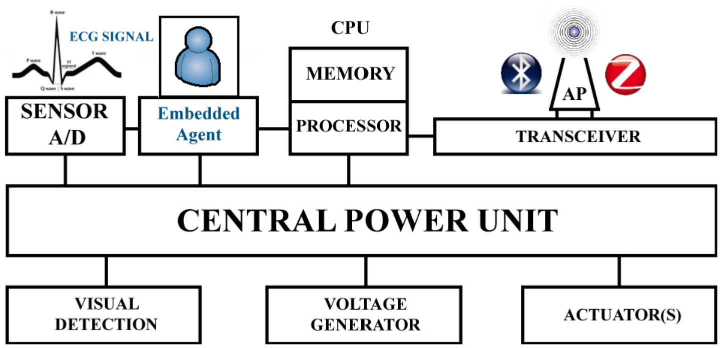


Fig. 7. Wireless sensor nod architecture and embedded agent location, [9]

4 Conclusion and Future Work

Power management for agent/multi-agent system is realized through embedded self monitored devices. With known battery voltage value (3.7 V) and measured battery current, the power consumption can be calculated. Installing the agents in the end user devices was done by programming the function that has the ability to alarm systems at a higher level. Then multi-agent manager system is activated, which further forwards the information to the distribution agent. If the alert level is very high, e.g. a heart attack occurred, the embedded master agent located in the embedded devices cloud is activated.

A possible future addition to the end device is the implementation of GPS receiver e.g. LEA6S uBlox. When the aforementioned level of alert is reached embedded master agent is able to locate the patient on the map.

Multi-agent system in the presented case, Fig. 3, is stored in CPX4 and CPX5 coordinators. When CPX5 is assigned in the network than it is a mobile multi-agent system who has an additional channel for data communications (Iridium Satellite). Multi-agent system is situated on the second layer and through the Internet or intranet and over the distribution agent it communicates with the most embedded master agent in the hierarchy. NI 9792 is a gateway and contains embedded device cloud integrated on its OS. This gateway has the option of additional ZigBee peripherals for its close sensors if there is a need for it.

The aim of this paper is to point out possible improvements of existing solutions and future development of advanced Holter devices, [10, 11, 12]. The biggest space for improvement is evident in the area of information transfer. Instead of off-line data which is at the moment stored on the SD card, the aim is to bridge the information automatically with SPI (Serial Peripheral Interface) to RS232 adapter interface XBee PRO S2B module, Fig. 8.

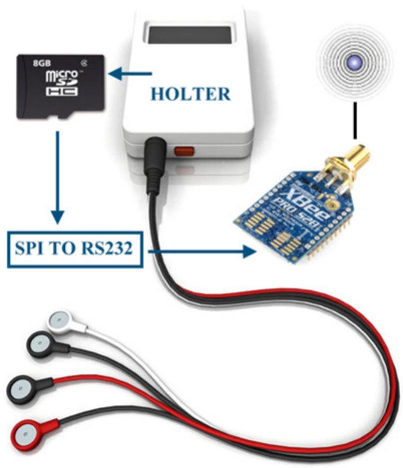


Fig. 8. Concept for future work (Holter on-line monitoring of ECG signal)

References

1. Biomedicinska elektronika, Školska knjiga, Zagreb, 382 str., Prof. dr. sc. Šantić, Ante (1995) ISBN 953-0-31637-2
2. Jin, Z., Sun, Y., Cheng, A.C.: Predicting cardiovascular disease from real-time electro cardiographic monitoring: An adaptive machine learning approach on a cell phone. IEEE Trans. Inf. Technol. Biomed. 14(3), 734–740 (2010), PMID:19964449, Epub April 12, 2010
3. Oresko, J.J., Duschl, H., Cheng, A.C.: A wearable smartphone-based platform for real-time cardiovascular disease detection via electrocardiogram processing. In: Conf. Proc. IEEE Eng. Med. Biol. Soc. 2010, pp. 5500–5503 (2010)
4. API and GUI from National Instruments: LabVIEW, `http://www.ni.com`
5. Wireless equipment. Digi., `http://www.digi.com`
6. Šoštarić, D., Vinko, D., Žagar, D., Samardžić, I.: Reliability of Welding Parameters Monitoring System. Strojarstvo (0562-1887) 52(2), 217–226 (2010)
7. Crossbow WSN platform, `http://www.xbow.com`
8. MikroElektronika Beograd, `http://www.mikroe.com`
9. Šoštarić, D., Vinko, D., Žagar, D.: JavaScript Virtual Web Page for Wireless Sensor Node under AVR Microcontroller Architecture. In: Wireless Mobile Networking Conference, Budapest (2010)
10. O'Hare, G.M.P., Marsh, D., Ruzzelli, A., Tynan, R.: Agents for Wireless Sensor Network Power Management. IEEE Xplore (2005) ISBN: 0-7695-2381-1
11. Ma, X., Qu, Y.: Reserch on Embedded Agent System Architecture. IEEE Xplore (2008) ISBN: 978-0-7695-3288-2
12. Montoya, A., Restrep, D.C., Ovalle, D.A.: Artificial Intelligence for Wireless Sensor Networks Enhancement. InTech (December 2010) ISBN 978-953-307-261-6

A Multi-Agent System for Dynamic Integrated Process Planning and Scheduling Using Heuristics

Luping Zhang[1], T.N. Wong[1], and Richard Y.K. Fung[2]

[1] Department of Industrial & Manufacturing Systems Engineering, The University of Hong Kong
[2] Department of Systems Engineering and Engineering Management, City University of Hong Kong

Abstract. Integrated process planning and scheduling (IPPS) is an NP-hard problem, the major research on the multi-agent system (MAS) based IPPS systems has focused on the establishment of negotiation protocols to accomplish the integration of process planning and scheduling. However, not much consideration has been paid to the dynamic factors of current manufacturing systems. In this paper, an MAS architecture is proposed to solve the dynamic IPPS problem with embedded heuristic algorithms. The proposed MAS system can be combined with a variety of heuristic methods to support dynamic process planning, scheduling and re-scheduling. As a result, the proposed MAS system for dynamic IPPS using heuristics possesses high flexibility, extensibility, and accessibility for manufacturing applications.

Keywords: Dynamic process planning, Jobshop scheduling, Multi-agent system, heuristics.

1 Introduction

Process planning and scheduling are two crucial functions which are usually performed sequentially in manufacturing systems as pre-production activities. In reality, the dynamic manufacturing environment involves uncertainties and disturbances such as machine breakdown and rush order. As a result, the process plan and schedule prepared in the planning stage may become infeasible or less efficient.

Integrated process planning and scheduling (IPPS) is to combine the process planning and scheduling functions in order to enhance the feasibility and optimality of the process plan and schedule [1]. Due to the fast growing complexity of products and processes, it is commonly believed that traditional exact algorithms are not suitable for solving the IPPS problem with reasonable efforts. An increasing number of approximate algorithms have been proposed for the IPPS problem domain, for instance, Genetic Algorithm (GA) [2], Simulated Annealing (SA) [3], and Particle Swarm Optimization (PSO) [4].

Multi-Agent Systems (MAS), as a promising distributed AI method, has prevailed gradually in the research of IPPS in recent years. Gu et al. [5] proposed a bidding-based MAS approach to integrate process planning and scheduling using autonomous machine agents. More recently, two negotiation-based MAS approaches MAN (Multi-Agent

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 309–318, 2012.

Negotiation) and HAN (Hybrid-based Agent Negotiation) for solving the IPPS problem were proposed by Wong et al [6]. One main consideration of the negotiation-based approach is the difficulty in determining the utility function and negotiation protocol. Whenever there are variations in products or machine capabilities, the utility function and negotiation strategies and protocols may have to be updated. That means, with the non-dynamic IPPS system, if changes happen in the manufacturing environment, everything has to start from scratch: new IPPS instance and negotiation protocols have to be constructed firstly so that the system can start all over again for new process plan and schedule. Instead of the purely negotiation-based MAS, a hybrid approach of MAS and the Ant Colony Optimization (ACO) was established later, in attempt to evaluate the effectiveness of the ACO in solving the IPPS problem in the MAS environment [7]. It was reported that the hybrid MAS-ACO approach could effectively solve the IPPS problems and provide opportunity for distributed computation of the ACO. In this current paper, a new MAS structure is proposed, which can not only handle uncertainties and disturbances such as machine breakdown and order rushing-in, but also provides an interface for heuristic methods to search near optimal solutions.

In the following sections, the dynamic IPPS problem is firstly described in section 2 with an AND/OR graph based representation. The main structure of the MAS system is described in section 3, together with main function agents and dynamic control procedures; a brief description of heuristics implementation is also given. Experiments and data are shown in section 4, where a concrete heuristic method ACO is implemented in the dynamic IPPS system and two kinds of environmental disturbance: order rushing-in and machine break-down are handled. Finally, section 5 presents the discussions and concluding remark.

2 Description of the Dynamic IPPS Problem

Generally, the IPPS model can be specified as **n** jobs to be processed on **m** machines with alternative process plans. Each job consists of a set of operations which are ordered by sequential relationships, and alternative machines are assigned to process each operation [8]. In this paper, the IPPS instance is illustrated with an **AND/OR** graph model **D=(O, A, B)** presented previously by Leung, et al. [7]. **O** is a set of nodes to represent operations belonging to each job. **A** is a set of directed arcs which define sequential relationships between operations. **B** is a set of undirected arcs representing possible heuristic trails for constructive meta-heuristics such as Ant algorithms. Figure 1 illustrates a sample **AND/OR** graph for a simple IPPS instance, where $\mathbf{O_s}$ and $\mathbf{O_f}$ are two dummy nodes representing the start and termination of all jobs. Job 2 has 2 alternative processes which is depicted with an **OR** relation. All undirected arcs except those connected to operation $\mathbf{O_4}$ are omitted to avoid confusion in the diagram. The dynamic IPPS means that the IPPS instance may be afflicted with environment uncertainties, which are typically machine breakdown or rush order coming in. It is necessary for the proposed system to possess the ability to sense changes in the environment and adjust itself accordingly.

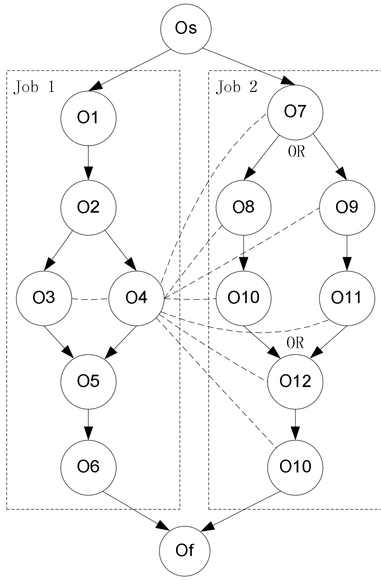


Fig. 1. A sample AND/OR graph

3 The MAS-Based Dynamic IPPS Architecture

The MAS architecture follows the theoretical model proposed by Van Dyke Parunak [9]. In accordance with this theoretical context, the MAS structure can be illustrated as follows.

$$\text{MAS} = \langle \text{Agents, Environment, Coupling} \rangle,$$
$$\text{Agents} = \{\text{Agent}_1, \ldots, \text{Agent}_n\},$$
$$\text{Agent}_i = \langle \text{State}_i, \text{Input}_i, \text{Output}_i, \text{Process}_i \rangle,$$
$$\text{Environment} = \langle \text{State}_e, \text{Process}_e \rangle.$$

The illustration above shows that the environment is an indispensible part of the MAS, and the interaction between agents and environment is also an important aspect. In the MAS-based dynamic IPPS system, we take great advantages of a global environment in the MAS architecture to hold all essential information for the IPPS instance. Various functional agents play their own roles in the global environment and coordinate with each other to dynamically generate the process plan and schedule. In the context of IPPS, this may involve a global objective or multi objectives such as minimizing the makespan, shortening the mean flow time, etc. Currently, our architecture comprises several kinds of functional agents, including the IPPS Supervisor Agent, model agents, Environment Maintenance Agent, Heuristic Supervisor Agent, and algorithm agents. Figure 2 depicts the MAS structure for dynamic IPPS, where the solid directed arcs represent the request actions between agents, and the dash directed arcs figures the information exchanges.

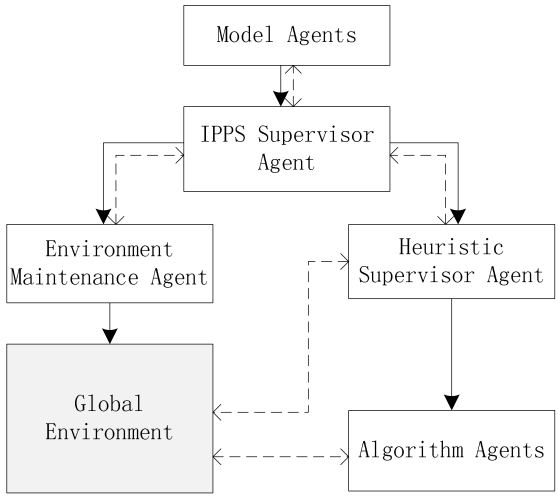


Fig. 2. The main structure of the dynamic IPPS system

The best benefit to implement heuristics for IPPS is to avoid the dependence on the static negotiation protocols and utility functions which may not be appropriate to cope with the dynamic process planning and scheduling requirement.

3.1 Main Function Agents

- Model Agent: the model agent is an abstract primitive agent type to represent the manufacturing elements in IPPS instance. The model agent can be instantiated to job agents, machine agents, etc. In practical application, model agents can monitor manufacturing elements such as jobs and machines in the manufacturing system. It can communicate with the agent supervising software agents and transmit information of uncertainties or disturbances to the IPPS system for process planning and scheduling from time to time.
- IPPS Supervisor Agent (ISA): it is a dominant agent which is responsible for the creation and control of supervisor agents for environment maintenance and heuristic control. Moreover, this agent also monitors the information sent from model agents and performs corresponding actions to adjust the IPPS system adaptively.
- Environment Maintenance Agent (EMA): the EMA is mainly responsible for the establishment and adjustment of the global environment of MAS when it receives a corresponding request from the ISA. For example, at the setup stage of the IPPS system when an initialization request comes, it reads the data stored in the data base and constructs an IPPS instance in the global environment. When the IPPS system is ongoing, it takes charge to adjust the IPPS instance and adapts it to the changes dynamically.

- Heuristic Supervisor Agent (HSA) and algorithm agents: the HSA is responsible for receiving heuristic request from the ISA, and then triggering the algorithm agents to search feasible solutions. The HSA is also responsible for the control of overall heuristic procedure, while algorithm agents are the concrete heuristic agents who perform the real heuristic activities to construct new solutions or evolve solutions to search neighborhood solutions.

3.2 Dynamic Response

In order to make it possible for all agents to communicate with each other, the dynamic IPPS system introduces the ontology to define the language and vocabulary which makes senses for all agents. Our system also introduces a Directory Facilitator (DF) technology as specified by the Foundation for Intelligent Physical Agents (FIPA) standard, in order to make the system more flexible and extensible. All agents in the dynamic IPPS system are registered in the DF with services they provide, for example, environment maintenance, heuristic supervision, etc. Hence, every kind of agents can be easily located and reached if needed.

An interaction protocol for the main dynamic procedure is represented in Figure 3.

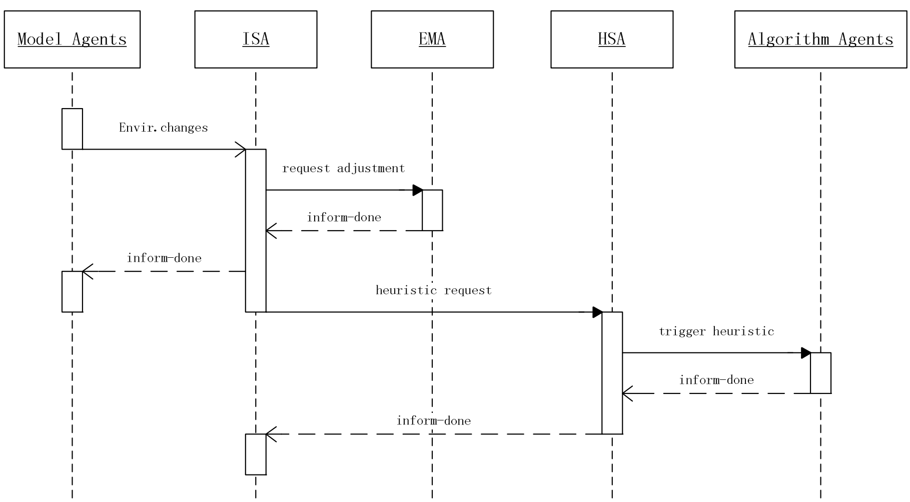


Fig. 3. The sequence diagram of the main dynamic procedure

To describe the main procedure, an example of a new rush order is assumed for explanation.

- When a job agent receives a rush order, it will notify the ISA about the detailed order information, including the jobs to accomplish the order, the quality of each jobs, and the delivery time.

- The ISA holds a queue of request for environment disturbances that are in need for handling. When the ISA catches an information of rush order sent from the job agent, it will generate an *update-order* request packaging the order information and pump it into the queue.
- At the end of each iteration when the HSA finishes the heuristic actions and sends back heuristic results, the ISA will check the unhandled changes queue. If some unhandled requests exit, the ISA will pump the request one by one and send it to the EMA for environment updating.
- The EMA catches the request of updating and identifies the update type. When it is an order rushing in, the EMA will acquire data of all newly requested jobs from the database and attach these additional jobs to the graph model in the global environment.
- After finishing the update of environment, the EMA will inform the ISAs about the update, such as new machines needed, etc.
- The ISA obtains the result of environment update and relays it to corresponding model agents for manufacturing control. For example, sometimes when a rush order comes into the system, new manufacturing resources are needed. This information will be sent back to model agents for new setting up.
- After that, the ISA will check the unhandled changes queue again. If no new unhandled request exists, the ISA will request the HSA starting a new iteration to search new solutions.
- Receiving the request to start a new iteration, the HSA runs the heuristic procedure and triggers the algorithm agents to search solutions. After new solutions have been generated, the HSA sends back these solutions and waits for the next request.
- The ISA catches all solutions generated by heuristic algorithms and calls for the solution handling module to process these solutions.

3.3 Heuristics Implementation

The proposed MAS architecture is designed to solve the IPPS problem with different heuristic algorithms. Heuristics can be classified into two categories: constructive heuristics and improvement heuristics [11]. The constructive heuristics usually construct new solutions at each iteration and skip steps for constructing and handling neighborhoods. Main steps of improvement heuristics can be illustrated as follows.

- Generate a population of solutions.
- Evaluate the affinity of each solution in the population, can choose a part of solutions of high affinity value for neighborhood construction and search.
- Construct neighborhoods and evaluate the affinity of neighborhoods.
- Involve the population for developing the superior and weeding out the inferior.

Ant algorithms and Greedy Randomized Adaptive Search procedure (GRASP) are typical constructive heuristics, whilst GA, PSO, etc. are typical improvement heuristics.

The proposed MAS architecture is able to accommodate either constructive or improvement heuristics. Currently, the GA-based and ACO-based heuristic approaches have been implemented for experimentation. For instance, if the ACO-based heuristic is adopted, the heuristic control agent (HCA) will direct the appropriate algorithm agents (implemented with ACO heuristic) to carry out the solution search. The focus of this paper is on the MAS architecture and hence details of the heuristic algorithms are not presented. What should be highlighted is that almost all heuristic activities are based on the global environment in the proposed system. The heuristic-related agents can acquire essential information for heuristic from the environment autonomously and exert impact on the global environment for evolution. Since heuristics are environment-based methods, this dynamic IPPS architecture can provide a compatible interface for the combination of heuristic methods to achieve dynamic process planning and scheduling.

4 Experiments

The proposed MAS structure for dynamic IPPS using heuristics has been implemented in JADE (Java Agent DEvelopment Framework). An ACO is implemented in our experiments to fulfill the heuristic functions.

The ACO is a constructive heuristic which can construct schedules through an autocatalytic process. Ants search paths on the graph which present solutions for the IPPS instance, and they deposit pheromone on visited paths so that there is a higher probability for coming ants to choose better paths. The transaction probability $p_k(u,v)$ of ants moving from node u to node v can be written as follows [7]:

$$p_k(u,v) = \begin{cases} \frac{[\tau(u,v)]^\alpha[\eta(u,v)]^\beta}{\sum_{w\in S_k}[\tau(u,w)]^\alpha[\eta(u,w)]^\beta} & \text{if } v \in S_k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$\eta(u,v) = \frac{C}{t_{v,m}} \quad (2)$$

$$\tau(u,v) = (1-\rho)\tau(u,v) + \Delta\tau_k(u,v) \quad (3)$$

$$\Delta\tau(u,v) = \begin{cases} \frac{Q}{L_k} & \text{if ant k transited from node u to v} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Where S_k is a set of possible nodes which ant can visit; $\tau(u,v)$ depict the pheromone intensity lying on path from node u to v; $\eta(u,v)$ denotes the heuristic desirability of ant from node u to v; α and β denote the weight parameters; $t_{v,m}$ is the process time of operation v on machine m; ρ is the evaporation rate of pheromone; L_k is the make-span by ant k; C and Q are both positive constants.

To start these experiments, coefficients are set as follows: number of ants=10; α=1; β=0.8; C=30; Q=250; ρ=0.95; original pheromone on all paths $\tau 0$=10. The algorithm is installed in the corresponding algorithm agent who keeps on searching new solutions. Experiments are performed on ThinkPad T410 laptop. The 18 parts 24 test bed problems proposed by Kim, Park and Ko [10] are adopted to build up the different IPPS instances. Two kinds of environmental changes are considered in our system, including machine breakdown and rush order.

To illustrate, problem No. 4 of the 24 test bed problems is chosen. This problem involves 6 jobs and altogether 109 operations. With regard to the scenario of machine breakdown, a software machine agent is implemented to simulate the behavior of a machine model agent. It is assumed that at one iteration when ants are searching for solution, the machine agent senses that machine No. 4 has broken down. The dynamic procedure of process planning and scheduling is then initiated. The HSA and algorithm agents are then requested to suspend, and the EMA is invoked by the ISA to disable machine No. 4 of the graph instance in the global environment. After that, the HSA and algorithm agents resume to search new solutions. Machine No. 4 cannot be reached any longer, thus no operations processed on machine No. 4 will be chosen and no tasks will be assigned to this machine until a new request to restore machine No. 4 has been issued by ISA. Figure 4 and Figure 5 depict the best schedule the ant algorithm can find before and after machine 4 breaks down respectively.

The makespan of best-so-far solutions corresponding to iterations is illustrated in Figure 5, where the machine M4 breaks down near the 500th iteration. Figure 5 shows that the number of iterations needed to find good solutions after machine breakdown is much smaller than that at the initial heuristic stage of the ant algorithm. It is mainly due to the fact that the ant algorithm does not start from the beginning but adjusts itself adaptively to search solutions in this dynamic IPPS system.

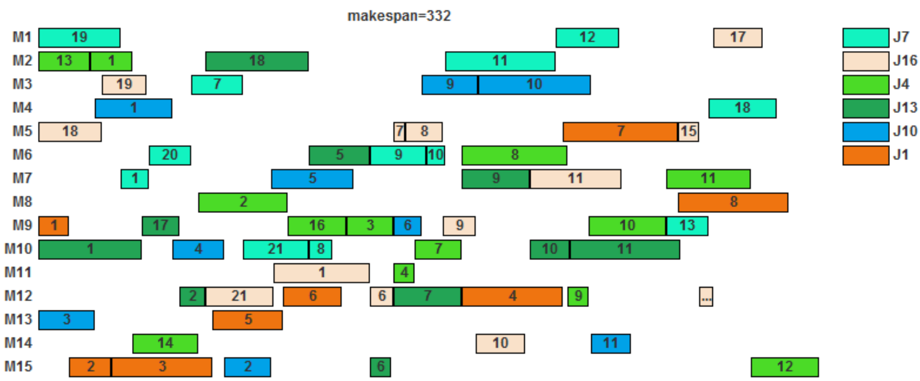


Fig. 4. The gantt chart for problem 4

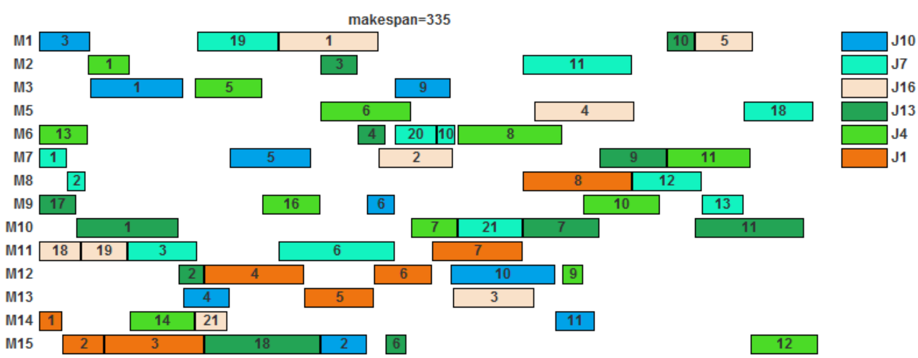


Fig. 5. The gantt chart for problem 4 after machine 4 breaks down

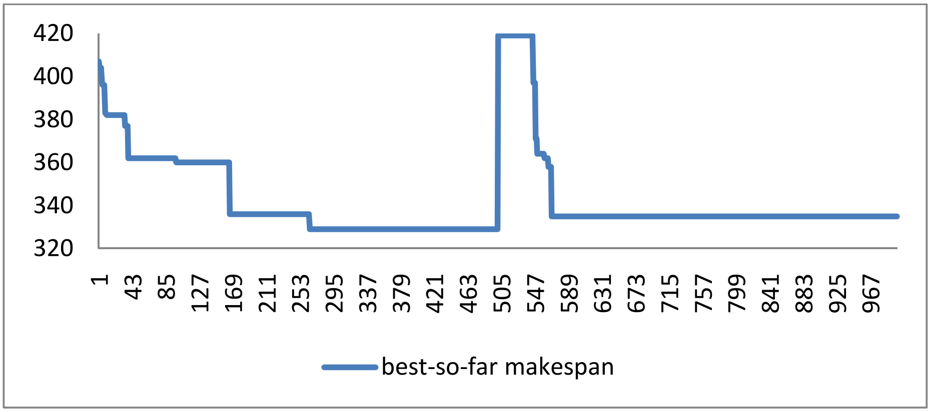


Fig. 6. The iterative makespan for the best-so-far solutions

In addition to machine breakdown, a software job agent has also been implemented to simulate the dynamic situation of rush order. Experiments show that the proposed system can also handle this dynamic situation easily. Since the data and results are very similar to those of the experiments on machine breakdown, these experiments are not described in detail in this paper.

5 Discussions and Concluding Remark

In this paper, a new MAS system is proposed for the dynamic IPPS problem, which combines the effectiveness of MAS and heuristics. This system can work together with various kinds of heuristic methods. The proposed dynamic IPPS system can handle the environmental changes like machine breakdown and rush order efficiently in our experiments and it works very well. The global environment in the MAS structure plays a dominant role, since it integrates functions of MAS framework and heuristic algorithms. Dynamic process planning and scheduling can be fulfilled with the interaction and negotiation between agents. However, no negotiation protocols are needed for heuristic algorithms to search near-optimal solutions. Hence, the adaptability and flexibility are greatly enhanced.

In addition to machine breakdown and rush order, other kinds of dynamic changes can also be handled. When new dynamic situations are considered, we just need to define the ontology, create new model agents and register them into the DF, and define the module for the EMA to handle these changes. The proposed IPPS structure can be extended easily.

Experiments have verified the feasibility of this new dynamic system. This system shows great flexibility and extensibility since multi dynamic situations can be considered and various kinds of heuristics can be integrated.

Acknowledgment. The work described in this paper is fully supported by a grant from the Research Grants Council of Hong Kong (Project Code HKU 718809E).

References

1. Chryssolouris, G., Chan, S., Suh, N.P.: An Integrated Approach to Process Planning and Scheduling. CIRP Annals - Manufacturing Technology 34(1), 413–417 (1985)
2. Morad, N., Zalzala, A.: Genetic algorithms in integrated process planning and scheduling. J. Intell. Manuf. 10(2), 169–179 (1999)
3. Li, W.D., McMahon, C.A.: A simulated annealing-based optimization approach for integrated process planning and scheduling. Int. J. Comput. Integ. M 20(1), 80–95 (2007)
4. Guo, Y.W., Li, W.D., Mileham, A.R., Owen, G.W.: Optimisation of integrated process planning and scheduling using a particle swarm optimisation approach. Int. J. Prod. Res. 47(14), 3775–3796 (2009)
5. Gu, P., Balasubramanian, S., Norrie, D.H.: Bidding-based process planning and scheduling in a multi-agent system. Computers & Industrial Engineering 32(2), 477–496 (1997)
6. Wong, T.N., Leung, C.W., Mak, K.L., Fung, R.Y.K.: Dynamic shopfloor scheduling in multi-agent manufacturing systems. Expert Syst. Appl. 31(3), 486–494 (2006)
7. Leung, C.W., Wong, T.N., Mak, K.L., Fung, R.Y.K.: Integrated process planning and scheduling by an agent-based ant colony optimization. Computers & Industrial Engineering 59(1), 166–180 (2010)
8. Guo, Y.W., Li, W.D., Mileham, A.R., Owen, G.W.: Applications of particle swarm optimisation in integrated process planning and scheduling. Robot Cim-Int. Manuf. 25(2), 280–288 (2009)
9. Van Dyke Parunak, H.: Go to the ant: Engineering principles from natural multi-agent systems. Annals of Operations Research 75, 69–101 (1997)
10. Kim, Y.K., Park, K., Ko, J.: A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling. Comput. Oper. Res. 30(8), 1151–1171 (2003)
11. Gendreau, M., Potvin, J.-Y.: Metaheuristics in Combinatorial Optimization. Annals of Operations Research 140(1), 189–213 (2005)

The Optimal Solution Attainment Rate of the Multiplexing Method

Yasuki Iizuka

School of Science, Tokai University, Kanagawa, Japan

Abstract. Distributed constraint optimization problems have attracted attention as a means of resolving distribution problems in multiagent environments. The author has already proposed a multiplex method targeting the improved efficiency of a distributed nondeterministic approximate algorithm for distributed constraint optimization problems. Since much of the computation time is used to transmit messages, improving efficiency using a multiplex computation of distributed approximate algorithms might be feasible, presuming that the computation time of each node or a small change in message length has no direct impact. Although it is usually impossible to guarantee that the approximation algorithm can obtain the optimal solution, the author managed to do so, using a theoretically determined multiplex method. In addition, the author shows the feasibility of an optimal solution attainment rate of 0.99 by an experiment using a Distributed Stochastic Search Algorithm.

1 Introduction

Distributed Constraint Optimization Problems (DCOP) are fundamental frameworks in distributed artificial intelligence and have recently attracted considerable attention [12,1]D As for DCOP, applications for traffic control and smart grids, etc. are expected. Several complete and approximate algorithms, [9,10,8] and [11,13] respectively, have been proposed as resolutions for DCOP. However, when tackling real-world problems, high efficiency algorithms are required [7], hence the purpose of this study is to increase the efficiency of distributed approximate algorithms. Although an approximation algorithm has a short computation time, obtaining the optimal solution cannot be guaranteed. In this paper, when an approximation algorithm is repeatedly performed, the probability that the optimal solution will be obtained is called the optimal solution attainment rate. This research targets an approximation algorithm with an optimal solution attainment rate of 0.99. Via the multiplexed execution of one kind of distributed approximation algorithm, despite the use of an approximation algorithm, it is shown that an optimal solution attainment rate can be arbitrarily established.

The next section describes an overview of DCOP. Details of the proposed multiplex method for DCOP and the result of a theoretical analysis of the expected effect using extreme value theory are described in Section three. The potential to establish an optimum solution attainment rate in the multiplex method

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 319–329, 2012.

proposed in this paper is discussed in Section four, while in Section five, the verification results of the multiplex method proposed in this paper through an example problem are described.

2 Distributed Constraint Optimization Problems (DCOP)

The definition of DCOP is described as follows [8,9]. A set of variables $x_1, x_2, ..., x_n$ exists, each of which is assigned a value taken from a finite and discrete domain $D_1, D_2, ..., D_n$, and each of which is also assigned to multiple agents $a_1, a_2, ..., a_m$. Constraints $c_{ij} : D_i \times D_j \to \{true, false\}$ are defined between x_i and x_j and a cost function $g_{ij} : D_i \times D_j \to \mathbb{R}^+$ exists for each constraint.

$$g_{ij}(x_i, x_j) = \begin{cases} v_{ij}^t, \; if \;\; c_{ij}(x_i, x_j) = true \\ v_{ij}^f, \; otherwise \end{cases}$$

Where, $v_{ij}^t < v_{ij}^f$. The agent a_k only has the following information: information about x_k , which is assigned to a_k, c_{ij}, which is a constraint of x_k, and the cost function g_{k*}. In this case, the purpose of DCOP is to obtain an assignment for variable $\mathcal{A}$ that minimizes the summation of the cost function $G(\mathcal{A}) = \sum g_{ij}(\mathcal{A})$ (Variable n is handled as $n = m$ here.)

In DCOP, an assignment $\mathcal{A}_o$ that offers the minimum $G(\mathcal{A}_o)$ amongst all possible assignments $\mathcal{A}$ is defined as an optimum solution. When an assignment $\mathcal{A}_p$ is $l = G(\mathcal{A}_p) - G(\mathcal{A}_o)$, $\mathcal{A}_p$ is defined as a solution of distance l from the optimal solution in this paper. Distance l is one of the measurement bases used to evaluate the quality of the solution. In DCOP, agents whose variables are associated by constraints solve problems by exchanging values of the variable through message transmission.

Well-known algorithms used to solve DCOP include ADOPT [9], DPOP [10], OptAPO [8], and distributed stochastic search algorithms (DSA) [13].

As for a complete algorithm, an optimum solution is guaranteed, despite the extended computing time. When using DCOP for real-world problems, particularly when solving problems involving robotics and sensor networks, problems must be solved in distributed environments with minimal computation resources [3,13]. To express complex issues, many variables are needed. Under such circumstances, seeking an optimum solution with a complete algorithm is not always the best method, and there is a greater need for a fast and efficient approximation method.

The author thus proposed a multiplex method targeting the improved efficiency of a distributed non-deterministic approximate algorithm for DCOP[6].

Related works such as algorithm portfolios exist [4], and a study of the parallel execution of approximate algorithms [2] one decade previously showed that not only could computation times be reduced but the quality of solutions could also be increased, a multiplex system for tree-based algorithms in DisCSP [5].

However, only the effect of the experiment was evaluated, despite engaging in theoretical consideration. Compared to these existing related works, the effect of multiplexing the non-deterministic approximation algorithm of DCOP, as proposed by the author, has the significant feature of being theoretically analyzed using extreme value theory.

In this paper the author theoretically shows the probability of obtaining an optimal solution that can be arbitrarily set according to this multiplex method.

3 Multiplex Method for Distributed Approximate Algorithms

This section explains the multiplex method of the distributed approximation algorithm as proposed by the author, and based on the following presumption:

Multiplexing, in this research, means the execution of several searches simultaneously without increasing computation resources. Because the problem was originally distributed for the DCOP, it is calculated by distributed agents and the use of the multiplex method means each agent's computation resource need not be increased. To avoid confusion with parallel processing, in which multiple computation resources are used, the term "multiplex" is used in this paper.

The computation time and quality of the solution of distributed approximate algorithms are subject to variation.

Most of the computation time in distributed algorithms is spent in message transmission [5]. Measures commonly used to evaluate distributed algorithms include ideal time complexities and communication complexities. Ideal time complexities refer to the overall number of message rounds, while each agent simultaneously exchanges and processes messages until the algorithm stops. Presupposing the adoption of this measure, if the number of messages remains constant, the processing time at each node and a slight increase in the length of each transmitted message will not impact significantly on the execution time of the distributed algorithm. Ideal time complexity will be used as the computation time of the algorithm in this paper.

3.1 Multiplexing Distributed Approximate Algorithms

When the results of a trial S conform to some kind of probability distribution, if the trial is repeated, and the minimum (or maximum) value of the repeated trials is selected as trial M, the latter conforms to a probability distribution different from trial S. For instance, if the scores are pointed to by a roll of the dice, the probability distribution becomes 1/6 each of $X = \{1, 2, 3, 4, 5, 6\}$, and the expected value μ_s of 3.5. If a dice is thrown m times, or a total of m dice are thrown simultaneously and the scores resulting from the minimum value of the dice are taken as trial M, the probability distribution will be non-uniform, and as the value of m rises, the expected value μ_m will become closer to 1.

If this fundamental is applied to distributed approximate algorithms, it may help improve their efficiency. Namely, if a distributed approximate algorithm is executed m times and the maximum or minimum value is selected as a result or if the algorithm is executed simultaneously in m layers and the maximum

```
Algorithm S
while(TerminationConditionIsNotMet)do
  mc := 0; /* Number of Messages */
  while(mc < #Neighbors) do
    neighborsStatus[mc] := Receive; mc++;
  endwhile
  x := NewValue(neighborsStatus[ ])
  Send(x)
endwhile
```

Fig. 1. Simple distributed iterative improvement algorithm $\mathcal{S}$

```
Algorithm Multiplexed-S
while(TerminationConditionIsNotMet)do
  mc := 0; /* Number of Messages */
  while(mc < #Neighbors) do
    neighborsStatus[mc][ ] := Receive; mc++;
  endwhile
  forall i ∈ m do /* for each plane */
    x[i] := NewValue(neighborsStatus[ ][i])
  end
  Send(x[ ])
endwhile
```

Fig. 2. Multiplexed distributed iterative improvement algorithm $Multiplexed\text{-}\mathcal{S}$

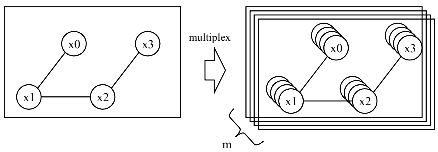


Fig. 3. Concept of the multiplexed algorithm

or minimum value is selected as a result, it may be feasible to gain a better computational performance.

In this paper, multiplexing the computations and messages at each respective node executing the distributed algorithm is considered, based on the above presumption, without increasing the number of messages.

Figure 1 is a simple example of distributed iterative improvement approximate algorithm $\mathcal{S}$ for DCOP. The agent executing this algorithm receives messages from the neighbor agents, and after determining its value x, informs the latter of this value. The procedure NewValue( ) is assumed to include nondeterministic elements. In this paper, the ordinary execution of an algorithm is considered as a plane. When multiple planes execute the algorithm simultaneously, it is defined here as multiplexing. Figure 2 shows $Multiplexed\text{-}\mathcal{S}$, which multiplexes the algorithm $\mathcal{S}$ of Fig. 1. The idea of multiplexing is shown in Fig. 3. Agents on m planes have independent values $x[i]$ and searches are performed independently on each plane when multiplexed, and though the length of the message is increased, there is no increase in the number of messages because in the messages exchanged with neighbor agents, the multiplexed value is exchanged in a single message.

In this paper, the number of planes simultaneously executed is defined as multiplicity and expressed as m.

3.2 Optimal Solution Attainment Rate and Multiplicity

When a distributed approximation algorithm is performed n times, suppose that it was m times the number of times that the optimal solution was obtained. At this time, $P_{opt} = n_{opt}/n$ is called the optimal solution attainment rate of that algorithm. There follows an argument concerning the optimal solution attainment rate at the time of multiplexed execution of the algorithm.

In [6], the author argued that a near-optimal solution was obtained by multiplexed execution of the algorithm. When multiplexed execution is performed, the mode of the probability distributions of the solution obtained diminishes, and likewise distribution. Subsequently, it is considered whether the attainment to the optimal solution can be guaranteed by raising the degree of multiplex. For example, suppose that there is a distributed approximation algorithm with a low attainment rate of the optimal solution. Can the optimal solution certainly be gained by multiplexed execution of this algorithm? In this section, the optimal attainment rate $P_{s\,opt}$ of the distributed approximation algorithm before multiplexing($m = 1$) is assumed to be known.

The distance from the optimal solution of the solution, as calculated by a certain distributed approximation algorithm, presumes that it becomes probability distributions, with the optimal solution the minimum of the latter. When the optimal solution attainment rate obtained by performing this distributed algorithm simply ($m = 1$) is $P_{s\,opt}$, the optimal solution attainment rate $P_{m\,opt}$ obtained by multiplexed execution is calculated by the following formula:

$$P_{m\,opt} = 1 - (1 - P_{s\,opt})^m \tag{1}$$

By using formula (1), the required multiplicity m can be calculated by establishing a target attainment rate relative to the optimal solution. For example, m in the case of an optimal solution attainment rate of 0.99, can be calculated as follows:

$$1 - (1 - P_{s\,opt})^m \geq 0.99 \tag{2}$$

thus

$$0.01 \geq (1 - P_{s\,opt})^m \tag{3}$$

By using common logarithms here, the following is obtained:

$$log 0.01 \geq log(1 - P_{s\,opt})^m \tag{4}$$

Because the left side is $log 10^{-2}$,

$$-2 \geq m\ log(1 - P_{s\,opt}) \tag{5}$$

Since $log(1 - P_{s\,opt})$ is negative, the conditions of the multiplicity m become the following formula:

$$m \geq \frac{-2}{log(1 - P_{s\,opt})} \tag{6}$$

It considers setting an optimal solution attainment rate to 0.99 by the multiplexed execution of the distributed approximation algorithm of the optimal solution attainment rate 0.2. Therefore, the required multiplicity m is calculable by the formula (6):

$$m \geq \frac{-2}{log(1-0.2)}$$

Since m is an integer, the condition is $m \geq 21$. Furthermore, the condition is $m \geq 31$ to set an optimal solution attainment rate to 0.999. Since a logarithm is used to calculate the degree of multiplex, when the conditions of the optimal solution attainment rate 0.9 are $m \geq m_1$, the conditions for an optimal solution attainment rate of 0.99 are $m \geq 2m_1$, and those for an optimal solution attainment rate of 0.999 are $m \geq 3m_1$.

Although an optimal solution attainment rate can be increased to 0.9999 etc., the attainment rate 1 to the optimal solution cannot be guaranteed. However, when raising the optimal solution attainment rate (when multiplicity m is enlarged), the distribution of the probability distributions of the distance from the optimal solution of that originally obtained diminishes, hence the distance from the optimal solution can become less than 1.

The above calculation shows the following, even if it is a distributed approximation algorithm with a low optimal solution attainment rate, namely that attainment is highly likely to be possible for the optimal solution using multiplexed execution. Although 0.99 was shown here, this rate can be freely established.

3.3 Selection of Efficient Multiplicity for Calculating the Optimal Solution

It was shown that an optimal solution attainment rate can be established by multiplexed execution of the distributed approximation algorithm by the argument in the previous paragraph. In this section, the case whereby the character of the distributed approximation algorithm not featuring multiplexed execution is an anytime algorithm[13] is considered. If the following two cases are compared, which will be efficient, this will involve the case of large multiplicity with a short execution round, and that of small multiplicity with a long execution round, respectively.

For distributed approximation algorithms which are not multiplexed, according to the computation time, monotonically decreasing distance from the optimal solution is assumed. Let l be the expected value of the distance from the optimal solution, r_e the number of rounds to the end of an algorithm and a and b constants. In algorithm DSt, which the author used for the experiment in [6], these relations have been approximated as in the following formula:

$$l = ar_e^{-b} \tag{7}$$

When not multiplexing an algorithm, the computation time is proportional to r_e.

First, we would like to denote the optimal solution attainment rate of this algorithm by the function of the number of rounds r_e to a stop. Henceforth, in this section, to simplify the model, the weight of violation of constraints is considered to be 1. Furthermore, it is assumed that the distribution of distance from the optimal solution obtained by the distributed approximation algorithm, with no multiplexed execution, is binomial. Parameters include the number of constraints n, and the probability $p(r_e)$ by which a certain constraint is not satisfied (when the number of rounds is r_e). At this time, the expected value can be described as follows as a function of r_e based on the binomial distribution definition:

$$\mu_s(r_e) = np(r_e) \tag{8}$$

Since this decreases by $l = ar_e^{-b}$, this can be described by the next formula.

$$np(r_e) = ar_e^{-b} \tag{9}$$

thus

$$p(r_e) = cr_e^{-b} \tag{10}$$

in which c is taken as a constant. The case where the number of violations of constraints in the optimal solution is 0 is considered for simplification. The probability of the calculated solution being an optimal solution can be expressed as a function of the number of rounds r_e to an algorithm stop.

$$\begin{aligned} P_{s\,opt}(r_e) &= (1 - p(r_e))^n \\ &= (1 - cr_e^{-b})^n \end{aligned} \tag{11}$$

Next, it asks for the minimum multiplicity for the optimal solution attainment rate 0.99 to be obtained as a function of r_e. Here, the formula (6) is made into an equals sign and the formula (11) is substituted, whereupon

$$m(r_e) = \frac{-2}{log(1 - (1 - cr_e^{-b})^n)} \tag{12}$$

By multiplex execution, we assumed that the computation time increases m times in the worst case. When multiplex execution of the algorithm applies and is made to stop at r_e round, the actual computation time becomes the following formula:

$$T_c(m) = r_e \times m \tag{13}$$

When formula (12) is substituted here, the computation time of the algorithm for multiplexed execution can be expressed as a function of the number of rounds r_e:

$$T_c(r_e) = r_e \times \frac{-2}{log(1 - (1 - cr_e^{-b})^n)} \tag{14}$$

The example result of numerical analysis using the conditions $0 \leq cr_e^{-b} \leq 1$ is shown in Fig. 4. Here, a figure is calculated by setting it to $c = 1, b = 0.9, n = 20,$

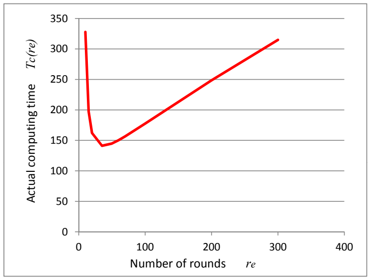


Fig. 4. Example of a numerical-analysis result of theoretical calculation

with r_e set as the horizontal axis and T_c as the vertical axis. The parameter set here is one of the examples. As shown in this figure, formula (14) may have a local minimum according to the conditions of the parameter. In other words, as a value peculiar to an algorithm, in order to calculate the optimal solution, the most efficient r_e and corresponding m will exist.

We would like to be cautious of the following: m calculated in this section is a value when observing the optimal solution attainment rate, while m calculated in [6] is a value for minimizing computation time. Since these m change based on the character of an algorithm, detailed analysis is taken as a future subject.

4 Verification by an Experiment to Determine the Optimal Solution Attainment Rate

Here, the optimal solution attainment rate by multiplex execution of the approximation algorithm for DCOP is verified by an experiment. The simple distributed approximation algorithm for DCOP is prepared, and an investigation performed to determine whether the optimal solution attainment rate becomes the theoretical value by the multiplexed execution of the same.

4.1 Experimental Conditions

In this experiment, as performed in the literature [9,11,13], distributed graph coloring problems are solved using a simulator, and the results were evaluated using the number of rounds in which messages are exchanged (ideal time complexity) and the optimal solution attainment rate. A weighted distributed constraint satisfaction problem that lacks a satisfying assignment must be chosen for the distributed graph coloring problem prepared for this experiment, and handled as a DCOP by expressing the condition's constraint satisfaction as a cost function of $\{0, 1, 2, 3, 4, 5\}$.

The distributed approximation algorithm DSA-B[13] was used for this experiment, the parameter of which was set to $p = 0.65$. In this experiment, since the purpose was to measure the effect of multiplex execution, the simulator was

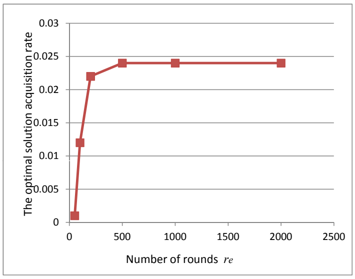


Fig. 5. The optimal solution attainment rate (in the case of $m = 1$) experimental result of an algorithm DSA-B

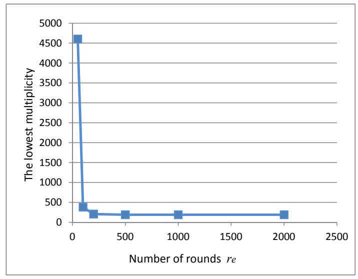


Fig. 6. The degree of minimum multiplex; computed in order to obtain an optimal solution attainment rate of 0.99 from an experimental result (Fig. 5)

stopped in arbitrary time and the solution of the plane having acquired the best value was chosen. Moreover, results were compared for the case where an algorithm was not multiplexed ($m = 1$), and that where it was ($m > 1$).

4.2 Experimental Results

Each of the following experiments involved the prepared problem being solved 1000 times, and the optimal solution attainment rate being investigated. First, the optimal solution attainment rate of the algorithm DSA without multiplexed execution, which is equivalent to formula (10), is shown if Figure 5.

Based on the result obtained here, the minimum multiplicity for obtaining the optimal solution attainment rate 0.99 was calculated, with the result shown in Figure 6. From these figures, the multiplicities for obtaining optimal solution attainment rates of 0.9 and 0.99 were respectively set up, and the multiplex execution of the algorithm was carried out. The result is shown in Table 1 and Figure 7.

As shown in Fig. 5, within the experimental range, the optimal solution attainment rate of the algorithm DSA used for this experiment was less than 0.1 and did not increase, even with extended computation time. However, optimal solution attainment rates of 0.9 and 0.99 were attained, as shown in Table 1, as a result of multiplexed execution of the algorithm. In other words, the optimal solution can be efficiently obtained by multiplex execution also about an approximation algorithm with a low optimal solution attainment rate. The effect of multiplex execution is considerable.

With regard to the theoretical value of multiplicity from the experimental result, the result of having applied the number of rounds and multiplicity is shown in Figure 8, which is equivalent to the actual measurement of the theoretical-analysis result of Fig. 4. Both Figs. 8 and 4 include a local minimum.

Table 1. The optimal solution attainment rate by multiplex execution (experimental result)

Experimental condition	The number of times of optimal solution acquisition	Optimal solution attainment rate
Target=0.9 Round=100 Multiplicity=191	940/1000	0.940
Target=0.99 Round=100 Multiplicity=382	996/1000	0.996

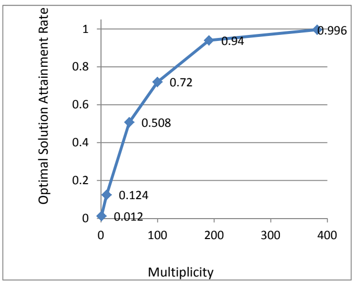


Fig. 7. The optimal solution attainment rate by multiplex execution

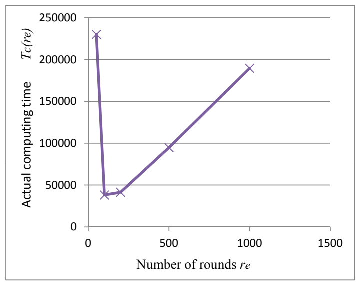


Fig. 8. Real computation time calculated from the experimental result (Figs. 5, 6)

5 Considerations

With regard to the optimal solution attainment rate at the time of multiplexed execution of the distributed approximation algorithm, a theoretical examination was performed in Section 3.2 and experimented in Section 4. Consequently, even when a distributed algorithm with an optimal solution attainment rate of less than 0.1 was used, it was shown that an optimal solution attainment rate of 0.99 etc. could be realized by multiplexed execution.

When multiplex execution is carried out, the distribution of the distance from the optimal solution diminishes [6]. Therefore, even if the optimal solution is not obtained, the obtained solution is one in which the distance from the optimal solution is negligible. In the experiment performed in section 4, which targeted an optimal solution attainment rate of 0.9, the optimal solution was obtained 940 times. For the remaining 60 times, the distance from the optimal solution was 1. Namely, the probability of becoming a distance of one or less from the optimal solution is 1.000. To measure accuracy to within 0.001, it is necessary to experiment by further increasing the number of measurements.

The fundamental view of this multiplexing solution is as follows, each plane is searched independently, the search is stopped independently, and an optimal solution is chosen, which can be said to resemble the concept of a genetic algorithm. Therefore, this method is considered applicable not only for DCOP but also the distributed search algorithm.

6 Conclusion

In this paper, when the multiplex execution of the distributed approximation algorithm for DCOP was carried out, it was shown that an optimal solution attainment rate could be set. Since a simple algorithm was applied for this experiment, further study of a method for the autonomous adjustment of multiplicity might improve the effectiveness of this study.

References

1. Calisti, M., Neagu, N.: Constraint satisfaction techniques and software agents. In: Agents and Constraints Workshop at AIIA 2004 (2004)
2. Falco, I.D., Balio, R.D., Tarantino, E., Vaccaro, R.: Improving search by incorporating evolution principles in parallel tabu search. In: 1994 IEEE Conference on Evolutionary Computation, pp. 823–828 (1994)
3. Fitzpatrick, S., Meertens, L.: An Experimental Assessment of a Stochastic, Anytime, Decentralized, Soft Colourer for Sparse Graphs. In: Steinhöfel, K. (ed.) SAGA 2001. LNCS, vol. 2264, pp. 49–64. Springer, Heidelberg (2001)
4. Gomes, C.P., Selman, B.: Algorithm portfolios. Artif. Intell. 126, 43–62 (2001)
5. Ringwelski, G., Hamadi, Y.: Boosting Distributed Constraint Satisfaction. In: van Beek, P. (ed.) CP 2005. LNCS, vol. 3709, pp. 549–562. Springer, Heidelberg (2005)
6. Iizuka, Y., Iizuka, K.: Exceeding the Efficiency of Distributed Approximate Algorithms Enabling by the Multiplexing Method. In: König, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) KES 2011, Part III. LNCS, vol. 6883, pp. 366–377. Springer, Heidelberg (2011)
7. Junges, R., Bazzan, A.L.C.: Evaluating the performance of dcop algorithms in a real world, dynamic problem. In: AAMAS 2008: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 599–606 (2008)
8. Mailler, R., Lesser, V.: Solving distributed constraint optimization problems using cooperative mediation. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 438–445. IEEE Computer Society (2004)
9. Modi, P.J., Shen, W.M., Tambe, M., Yokoo, M.: Adopt: asynchronous distributed constraint optimization with quality guarantees. Artif. Intell. (161), 149–180 (2005)
10. Petcu, A., Faltings, B.: A scalable method for multiagent constraint optimization. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 266–271 (2005), http://jmvidal.cse.sc.edu/library/petcu05a.pdf
11. Yeoh, W., Koenig, S., Felner, A.: Idb-adopt: A depth-first search dcop algorithm. In: Eighth International Workshop on Distributed Constraint Reasoning, in Twentieth International Joint Conference on Artificial Intelligence, pp. 56–70 (2007)
12. Yokoo, M., Hirayama, K.: Algorithms for distributed constraint satisfaction: A review. In: Autonomous Agents and Multi-Agent Systems, vol. 3, pp. 198–212 (2000)
13. Zhang, W., Wang, G., Xing, Z., Wittenburg, L.: Distributed stochastic search and distributed breakout: properties, comparison and applications to constraint optimization problems in sensor networks. Artif. Intell. (161), 55–87 (2005)

Modeling Secure Mobile Agent Systems

Molka Rekik, Slim Kallel, Monia Loulou, and Ahmed Hadj Kacem

ReDCAD Laboratory, University of Sfax
B.P. 1173, 3038 Sfax, Tunisia
molka.rekik@redcad.org,
{slim.kallel,monia.loulou,ahmed.hadjkacem}@fsegs.rnu.tn

Abstract. We propose a MDA based approach for designing secure mobile agent systems. First, we define a meta-model which extends the UML deployment diagram by concepts related to the security and mobility of multi-agent systems. We propose also a UML profile as an implementation of this meta-model. Second, we project the application model into AGLETS-specific model, which describes the main functionalities of the application deployed on AGLETS as a mobile agent platform.

1 Introduction

The mobile agent technology is adequate to the development of several complex applications due to the ability of agents to achieve their goals in flexible manner at different distributed sites. However, these agents as well as their execution systems require a security level that varies according to the application.

The complexity of the basic concepts related to the mobility and the security emerges some difficulties in the development of mobile agent systems. For this reason, the design and the development of such applications must be rigorous and assisted. Some research approaches have been proposed [1–4] to master the complexity of the mobility and the security of multi-agent systems.

However, several limitations have been identified: *First*, the specification of the security requirements is limited only to control mobile agent behaviors and their access resources. This lack of investigation, is justified by the double complexity bound, on one hand, to the variety of the concepts for expressing security policies in mobile agent system and on the other hand to the richness of the concepts which describe mobility in this system. *Second*, some approaches, like [4] are based on formal methods for specifying and verifying secure mobile agent systems. Despite the advantages of formal techniques, these approaches are quite complex and their use requires a specialist. *Third*, not all steps in the process of implementing security properties are covered; these properties are not specified in the early phases of software development but rather added later, which negatively affects the application code's quality. Therefore, these approaches do not support a refinement technique between the different phases.

To overcome all previous limitations, we propose an approach MDS4MAS (Model Driven Security for Mobile Agent System) which adoptes model-driven architecture for developing secure mobile agent systems. MDS4MAS is inspired

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 330–339, 2012.

by the work of Basin et al.[5], which proposed a generic approach for building secure information systems. They use the MDA approach and combine design modeling and security modeling languages to specify the secure system along with its access control requirements. This work has motivated ours for specifying security properties in mobile agent system.

Our approach is composed of two main phases: The first phase corresponds to the PIM level of the MDA approach. We extend the UML deployment diagram by defining a new meta-model for specifying new concepts required to model the security and mobility of agent systems. We propose also two UML profiles as an implementation of this meta-model. The first one *MobilityProfile* extends UML in order to model a mobile agent systems. While, the second profile *SecurityProfile* consists of modeling security aspects of such systems. In addition to a set of stereotypes, this profile defines a set of OCL constraints to specify the security constraints. The second phase, know as PSM level in MDA approach, consists of generating a new model as a result of projecting the application model in a specific platform. We define a set of transformation rules on the proposed model to automatically generate an AGLETS specific model. This model describes the main functionalities of the application using the AGLETS concepts.

Our approach offers a better conceive of the complexity of mobile agents and their needs for security. It supports also most of security concepts related to mobile agent system. Based on MDA approach and on the refinement techniques, our approach covers most of the development process of such systems and it can be applied automatically in other mobile agent platforms.

The remainder of the paper is organized as follows: Section 2 explains the proposed meta-model and profile for modeling of the secure mobile agent systems. In Section 3, we present the Aglets specific modeling. Section 4 presents the validation of our approach and Section 5 reports on related work. Section 6 concludes the paper and discusses areas of future work.

2 Modeling Secure Mobile Agent Systems

In this section, we detail the proposed meta-model and the defined profiles for modeling secure mobile agent system.

2.1 The MDS4MAS Meta-Model

We propose a new meta-model, which supports the essential concepts for specifying secure mobile agent systems. As shown in Fig. 2.1, our MDS4MAS meta-model is composed in two packages. The first one describes the mobility concepts while the second one is interested in the security of mobile agent systems.

The mobile agent systems are marked by a lack of consensus about their fundamental concepts and the relations between them. In fact, several definitions have been proposed about their basic concepts. These definitions depend on the point of view of the authors and the application domain of the mobile agent. According to the study of several mobile systems, we have proposed in our previous

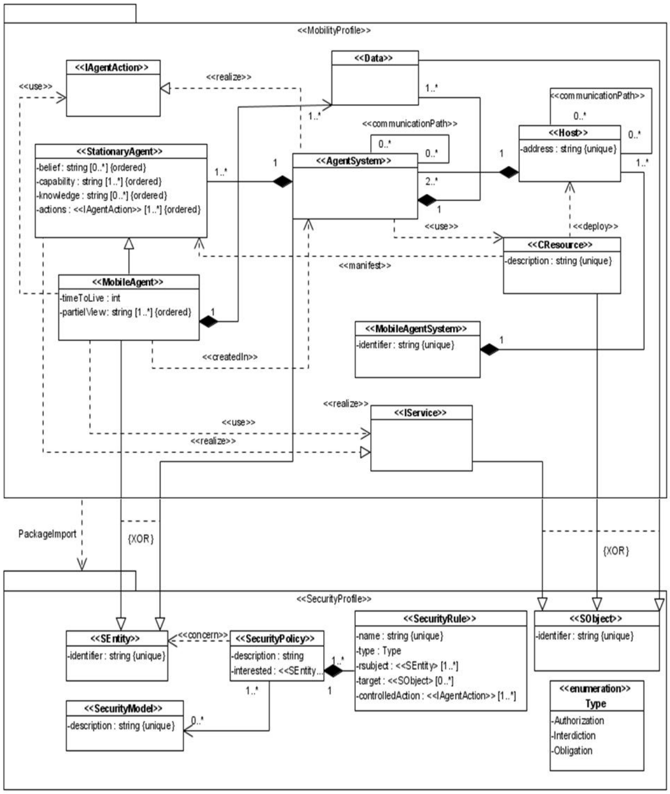


Fig. 1. MDS4MAS meta-model for secure mobile agent systems

work [4], a conceptual model that rigorously specifies the key concepts of secure mobile agent systems and unifies their representations independently of the specific application domain. On this base, we propose the following MDS4MAS meta-model.

A Mobile Agent System (denoted in our Meta-model by *MobileAgentSystem*) is a computer network composed of a set of interconnected host machines *Host*. Each one, has a unique name and described by a set of computing resources *CResource* which design its hardware features. A host machine can contain one or more agent systems *AgentSystem*. An Agent System is a set of agent evolving within an environment. This latter offers the basic functionality for mobile agent execution. Indeed it ensures agent creation and initialization, reception of incoming agents, communication (local or distant) between agents, access to resources, agent migration, etc. These control services *IService* and others of application will be ensured by service agents. Every agent system is described with a unique name, a localization (the host on which it is installed), a set of offered services, a set of stationary agents *StationaryAgent* which must be able to accomplish the services offered by the system and a set of resources put on its disposal. These resources may change according to the requirements of the incoming agents. When the agent leaves the system, some resources will be liberated.

A Mobile agent *MobileAgent* is an active entity capable to migrate from one site to another in order to get nearer to the required resources and services to accomplish properly its goals. A mobile agent can be specified as a stationary agent which should have some other attributes to express its mobility. Thus, a mobile agent should be identified by a name defined at its creation. The mobile agent acts according to its believes, its capability and its knowledge in order to achieve actions *IAgentAction* that are affected to him. It defines its new localization according to its requirements in terms of resources and services and according to its partial view (i.e.; *partitialView* is an attribute in the meta-class *MobileAgent* of inter-hosts connections).

Both agent system and mobile agent should have well defined security policy with the aim to screen the incoming agents and/or adversary agent system respectively adversary mobile agent and hosting agent system. Thus, a secure entity *Entity* can be either a mobile agent or an agent system. Mobile agents and Agent systems aim to protect their secure objects denoted by *SObject*. A secure object may be either data *Data*, or service or computing resource *CResource*.

A security policy *SecurityPolicy* regroups a set of security rules *SecurityRule*. Each one is described by : a type of the security rule *Type*, the secure entity concerned with the security rule *Interested*, the subject entity *RSubject* on which we apply it, the target object *Target* and a no empty set of actions *ControlledAction* to be enforced by the rule to reach the desired behavior.

2.2 The MDS4MAS Profile

As an implementation of the MDS4MAS meta-model, we propose an UML profile as an extension of the UML2.0 deployment diagram for modeling secure mobile agent systems. This profile defines all previous cited concepts in our meta-model as UML elements through definition of stereotypes for each meta-class. In addition, our profile contains a set of OCL constraints to impose some restrictions on the defined stereotypes. All stereotypes and their constraints are collected in a UML profile as schematically shown in two parts. The first part presents the mobility concepts of Mobile agent systems. For example in the first part, the stationary and the mobile agent extend the meta-class *component*, while the agent system and the mobile agent system extend respectively the meta-class *Execution environment* and *node*.

The second part defines the previous security concepts using UML. As an example, the *SecurityModel*, *SecurityPolicy* and *SecurityRule* are defined as *class* and the *SObject* extends the meta-class *Artifact* and *Interface*. Concerning the *SecurityRule* class, it will be described at the modeling level. Since, the designer should define, principally, the type of this rule. As well as, he should specify the security entity concerned, the object to protect and the action to control. In order to give more completeness to the definition of this constraint, it should be expressed in a formal and standardized way by using the OCL langage.

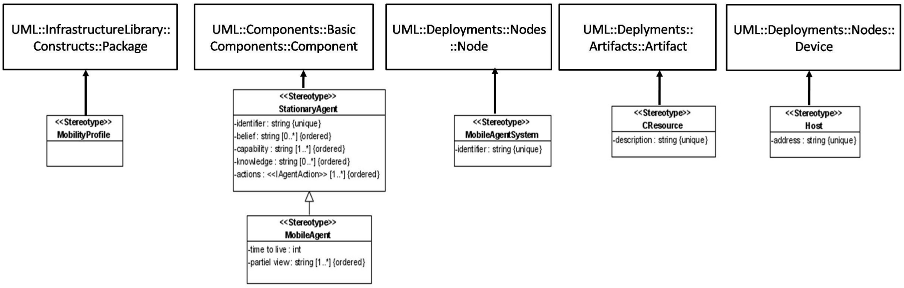


Fig. 2. MDS4MAS UML profile (Part1)

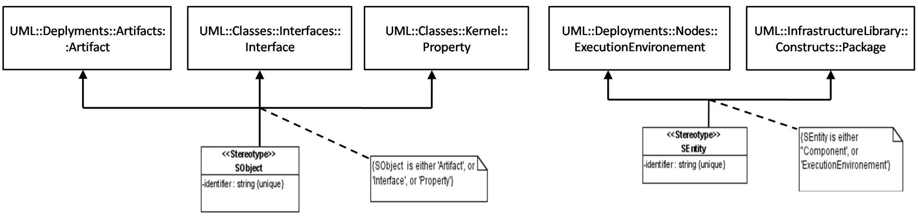


Fig. 3. MDS4MAS UML profile (Part2)

3 AGLETS Specific Modeling

According to the MDA approach, the next step corresponds to the transformation of application model into platform specific model which contains all the necessary information about implementation. So, we need essentially to specify a deployment platform for mobile agents.

We select the AGLETS platform [6] for three main raisons: *First*, AGLETS is virtually compatible with all operating systems, open source, free, light and well documented. *Second*, it offers an environment for programming mobile Java objects which react like mobile agents that can move from machine to another in a computer network by using the ATP (i.e., Agent Transfer Protocol). *Third*, Several concepts of the AGLETS platform are defined in our meta-model. In the following, we present the most important concepts of the AGLETS platform:

- **Aglet** is a mobile agent, its creation requires the definition of a class that extends the Aglet class and some methods which cover the important events in its life cycle, such as: onCreation, OnArrival, etc. The aglets are Java objects which have in more faculty of mobility.
- **AgletID** represents the unique identifier for an Aglet.
- **AgletContext** represents the environment of execution which manages the life cycles of the aglets by offering them services and protecting the host against malicious aglets.
- **Message** allows the synchronous communication and the exchange of information between agent (aglets).

The programming of the agents under AGLETS is in close connection with the life cycle of the agent. Indeed, an aglet is created in a context and at this time an identifier is assigned. It can be disabled for a period of life. The agent can also be transferred from a context to another with the ability to recover its original context. Its life cycle ends in its release or its destruction. The communication between aglets is based on the exchange of objects of Message class.

We proposed an UML profile partially describing the AGLETS platform. It allows to automatically project the application model into AGLETS specific model. As shown in Figure 4, this profile represents only the previous cited AGLETS concepts as stereotypes defined by specific meta-classes of UML meta-model. *Aglet*, *AgletID*, and *Message* extend the meta-class class, while *AgletContext* is represented as an extension of an *Interface*.

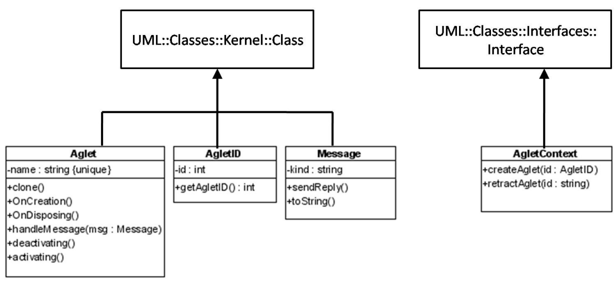


Fig. 4. UML profile for a partial description of AGLETS platform

We use Atlas Transformation Language (ATL) [7] for automatically translated the application model into AGLETS specific model. We start by defining the mapping between the concepts at meta-models level. Table 1 describes the mapping suggested between our MDS4MAS meta-model and the meta-model of AGLETS. These matches lead to determine transformation rules needed to be applied on all application models in conformity to the application meta-model.

Table 1. Mapping between application model and Aglets model

Name of mapping	Concept in PIM level	Concept in PSM level
M2M	MobilityProfile	AgletProfile
AgM2Aglet	StationaryAgent ou MobileAgent	Aglet
AgM2AgletID	name	AgletID
Message	nothing	Message
AgS2AgletContext	AgentSystem	AgletContext
AgAc2O	IAgentAction	nothing

We take for example the transformation rule named *AgM2Aglet*, which connects the element stereotyped Agent whatever *StationaryAgent* or *MobileAgent* of application model with the element stereotyped *Aglet* of AGLETS specific model, by keeping the same name and attributes. We call a set of lazy rules to add some attributes and operations which are necessary for the implementation of our application. These attributes and methods are specific to AGLETS.

Listing 1.1. Part of the ATL transformation rules

```
1  rule AgM2Aglet {
2      from s : MM!"uml::Component"
3      to t: MM1!"uml::Class" (
4           name <- s.name,
5           ownedAttribute <- s.ownedAttribute
6           ownedAttribute <- s.ownedAttribute->collect(c|thisModule.AgM2AgletID(c))
7            )
8      do {
9          thisModule.package.packagedElement <- t;
10         t.applyStereotype(thisModule.getStereotype('Aglet'));
11         if (s.hasStereotype('MobileAgent')) {
12              t.ownedAttribute
13                      <- thisModule.itinerary(MM!MobileAgent.allInstances().first());
14              t.ownedOperation
15                      <- thisModule.AgAc2O2(MM!"uml::Interface".allInstances()
16                      -> select(e | e.oclIsTypeOf(MM!"uml::Interface"))-> at(2));
17              }
18       }
19 }
```

This rule copies both the name and the attributes of the source UML component to the target UML Class (line 1 to line 7) and connects them to its UML Class (line 10). Then, we should accord to this class, the *Aglet* stereotype (line 11). Also, this rule adds the set of all the attributes and operations necessary of execution of the AGLETS platform. If the source UML component is stereotyped *MobileAgent* (line 11), we need add for example the itinerary attribute and atHome, onCreation methods (lines 12 – 19).

4 Validation of Our Approach

We implemented a graphical editor as an Eclipse plug-in, so that the designer can easily model his secure mobile agent system based on the defined concepts.

In the following, we briefly illustrate our case study E-Commerce and we define the corresponding model, the transformation rules to generate an AGLETS specific model. Figure 5 presents the modelling of the E-commerce system as a case study, which is composed by set of stationary and mobile agents. A mobile agent *Buyer* is created in a system agents *SellerSystem1*. This agent can move to another agent systems to buy computer resource *Printers*. This agent can move to *SellerSystem2* where exists the *Seller* presented as a stationary agent, responsible for the action to sell the *Printer* resource.

As an example, we defined the security rule SR1, which prohibits the mobile agent Buyer to buy the computer resource Printer with the stationary agent

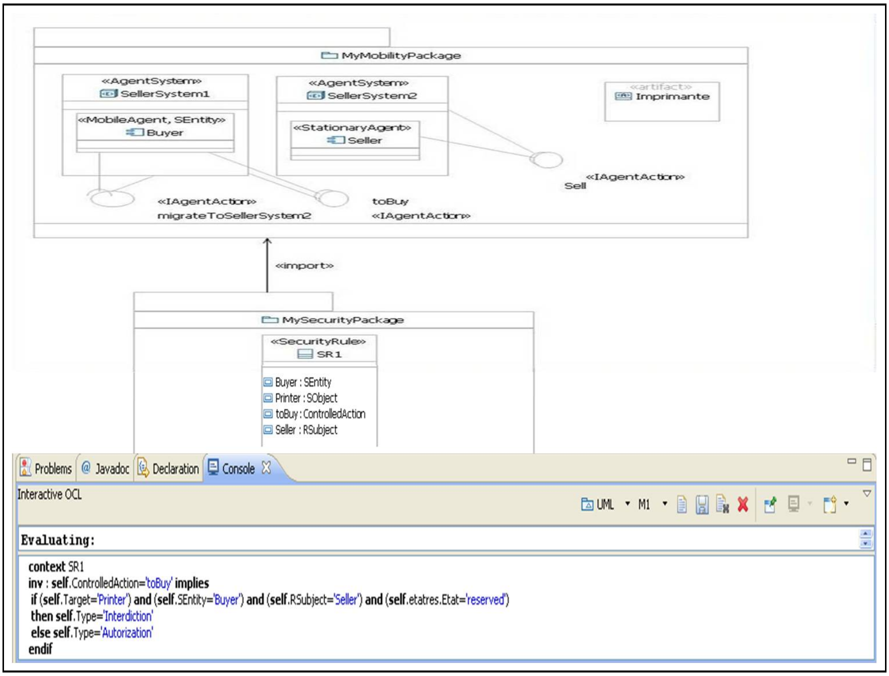


Fig. 5. A model of the E-commerce system

Seller if this Printer is reserved to be sold. As shown in the figure 5, the designer should, firstly, describe his security rule Class. SR1 presents her name, Buyer defines the Security Entity, Seller determines the security subject, Printer presents the target to protect and toBuy defines the action to control. Secondly, he should specify in the shutter console of our MDS4MAS editor, especially in the tab Interactive OCL, his security constraint by using the OCL language.

According to our approach, we applied the defined transformation rules to translate the previous platform-independent model (Figure 5) to the corresponding AGLETS specific model, but without supporting security concerns. This model, shown in Figure 6, is more detailed and near to the technical solution, since it be automatically translated after that to the AGLETS application code.

As already explained in the previous section, the transformation rules automatically generates further methods and attributes for the mobile agent class. For example, in the *Buyer* class, only the method *toBuy* is translated from the application model, all others methods (e.g., athome, startTrip) are automatically created to satisfy the AGLETS requirements for developing mobile agent.

5 Related Work

Several works have been proposed for modeling secure mobile agent systems.

Elhum et al. [8] present a security communication model *SAGLET* basing on existing Aglets architecture. They involve protecting the state of the aglets

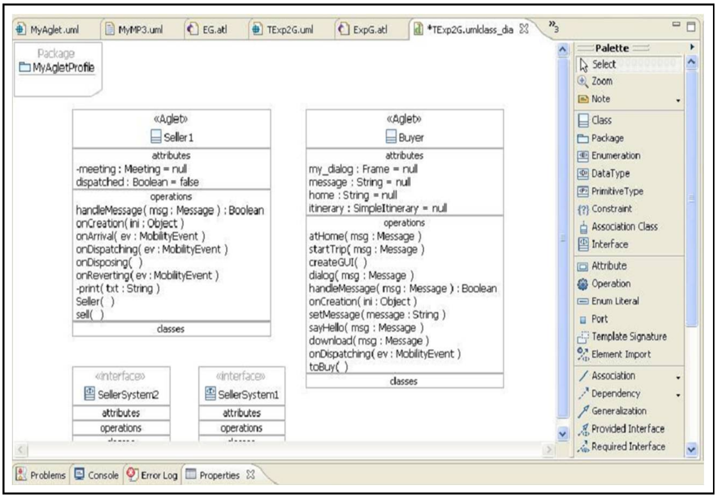


Fig. 6. Platform-specific model of AGLETS

and their malicious activities. They propose a new service agent along with a specific policy. This service agent allows the authentication of the visiting agents, the control of the communication between service agent and visitor agent, and allocates resource to agent according to the defined policy. This work focuses only on the modeling of the security of the communication between agents and does not provide a framework for specifying the security of the agents and the agent systems as defined in ours. In addition, this work is specific to Aglets and cannot be implemented on several platforms.

In [9], the authors propose a secured aglet server (SAS), which ensure three security functionalities. This tahiti server ensures: (i) the security of the communication between agents by implementing the SSL using the Java secure socket extension, (ii) the control of agent resources using an a MonitorAglet, which is responsible to track and verify the resources used by particular aglets. (iii) the integrity and reliability of data agent. Contrary to our approach, this work does not provide a hight level specification of the provided security functionalities, which negatively affects the quality of the application code.

Jarraya et al. [1] propose a new multi-agent development method, named MDAD (Model Driven Agent Development). In the PIM level, the authors defined a library of meta-models to describe the multi-agent system independently of the development platforms. At the PSM Level, the authors represented the components of a Multi-agent system in terms of the INAF development platform. This work does not offer to the developer a graphical tool to design PIM and/or PSM models. Moreover, the authors are interested only to model the concepts related to agent, without however study the security aspects.

Fink et al. [2], present a MDA approach for specifying of the control access policies in distributed systems. The authors supported the VBAC (View-based Access Control) model of control access. In the meta-modeling phase, the VBAC-PIM-MM and VBAC-PSM-MM are expressed by the MOF language enriched by UML profiles. Then, they establish the necessary connections for the passage of VBAC-PIM-MM and the VBAC-EJB-MM. In the modeling phase, the authors developed the VBAC-PIM, for a management conference application, which will be compiled into a VBAC-EJB. This work, is quite complex because the authors have proposed a heavy extension in which they have changed the MOF. In addition, they adopted the VBAC model so they consider only the control access policy. Moreover, the generation of the VBAC-EJB model from the VBAC-PIM does not have explained due to they did not use a transformation language.

In our work, we benefit from presented works while remedying their limitations. We adopted MDA approach while considering a high abstraction level to model secure mobile agent systems. Further, our work can be applyed on several platforms. It is sufficient to define the corresponding transformation rules.

6 Conclusion and Future Work

We presented a model-driven approach for the modeling secure mobile agent systems. We proposed a meta-model to represent the mobility and the security concepts related to multi-agent systems. We proposed also a UML profile as an implementation of this meta-model. In addition, we proposed a set of rules to transform the platform independent model to AGLETS-specific model.

As future work, we plan to apply our approach for generating Jade-specific model from the platform-independent model. We plan also to profit of the advantages of Aspect-oriented programming by generating AspectJ aspects to enforce at runtime the security constraints specified using OCL as defined in [10].

References

1. Jarraya, T., Guessoum, Z.: Towards a Model Driven Process for Multi-Agent System. In: Burkhard, H.-D., Lindemann, G., Verbrugge, R., Varga, L.Z. (eds.) CEEMAS 2007. LNCS (LNAI), vol. 4696, pp. 256–265. Springer, Heidelberg (2007)
2. Fink, T., Koch, M., Pauls, K.: An MDA approach to Access Control Specifications Using MOF and UML Profiles. In: Proc. of the 1st VODCA, pp. 161–179 (2004)
3. Bryce, C.B.: A Security Framework for a Mobile Agent System. In: Cuppens, F., Deswarte, Y., Gollmann, D., Waidner, M. (eds.) ESORICS 2000. LNCS, vol. 1895, pp. 273–290. Springer, Heidelberg (2000)
4. Loulou, M., Jmaiel, M., Mosbah, M.: Dynamic Security Framework for Mobile Agent Systems: specification, verification and enforcement. IJICS, 321–336 (2009)
5. Basin, D., Doser, J., Lodderstedt, T.: Model Driven Security: From UML Models to Access Control Infrastructures. ACM TOSEM 15, 39–91 (2006)
6. IBM: Aglets (1996), http://www.trl.ibm.com/aglets/
7. Jouault, F., Kurtev, I.: Transforming Models with ATL. In: Bruel, J.-M. (ed.) MoDELS 2005. LNCS, vol. 3844, pp. 128–138. Springer, Heidelberg (2006)
8. Nusrat, E., Ahmed, A.S., Rahman, G.M., Jamal, L.: SAGLET- Secure Agent Communication Model. In: Proc. of 11th ICCIT 2008, pp. 371–375. IEEE (2008)
9. Jean, E., Jiao, Y., Hurson, A.R., Potok, T.E.: SAS: A Secure Aglet Server. In: Proc. of Computer Security Conference (2007)
10. Kallel, S., Kacem, M.H., Jmaiel, M.: Modeling and enforcing invariants of dynamic software architectures. Software and System Modeling 11, 127–149 (2012)

An Influence of Random Number Generation Function to Multiagent Systems

Dominik Vymětal, Marek Spišák, and Roman Šperka

Silesian University in Opava, School of Business Administration in Karviná, Department of Informatics, Univerzitní nám. 1934/3a, 733 40, Karviná, Czech Republic
{vymetal,spisak,sperka}@opf.slu.cz

Abstract. The paper deals with modeling and simulation of business processes. A multiagent system was implemented as a tool to manage the simulation. Multiagent systems often operate with random (respectively pseudorandom) generated parameters in order to represent unpredictable phenomena. The aiml of the paper is to show the influence of different random number generation functions to the real multiagent system outputs. It is obvious, that outputs of the multiagent system simulation differs from turn to turn, but the motivation was to find, if the differences are significant. An accurate number of agents with the same parameters were used for each case, with different kinds of randomness while generating agent's internal state attributes. The results obtained show that using inappropriate random number generation function leads to significant output data distortion, so the generation function selection must be done very carefully.

Keywords: random, pseudorandom, generation, function, multiagent, simulation, modeling, business process.

1 Introduction

Multiagent System which is currently being implemented shall be a tool for simulation of business processes. When finished, it shall cover the whole production company structure from supply of the material, through the production process, up to the selling and shipment, which is supported by marketing department. The governance of the whole system is in power of management.

The simulations we experiment with could be described as agent-based simulations [Macal and North, (2005)]; [Yan et all, (2001)] of business processes. Business process is an activity adding the value to the company. Usual business process simulation approaches are based on the statistical calculation [e.g. Scheer and Nuttgens, (2000)]. But only several problems can be identified while using this method. There are a lot of other influences that cannot be captured by using any business process model (e.g. the effects of the collaboration of business process participants or their communication, experience level, cultural or social factors). This method has only limited capabilities of visual presentation while running simulation. Finally, we do not actually see the participants of business process dealing with each other.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 340–349, 2012.

Agent-based simulations and their usage in a simulation of a company can bring several crucial advantages [De Snoo, (2005)]; [Jennings et all, (2000)]; [Moreno et all, (2003)]; [Sierhuis, (2001)], and can overcome some of the problems identified in the previous text. Agents that represent business process participants are more accordant with people and issues like communication, coordination or cooperation. It is possible to involve the disturbances of the environment into the simulation with agents. All of these issues are the basic characteristics of a multi-agent system (MAS).

Our simulation methodology is based on the idea that the company could be presented as a control loop, where the market conditions as well as customers behavior are seen as an external part of the modeled system while the internal company behavior is subject to the simulation [e.g. Wolf, (2006)]. We simulate core business processes of a business company like selling the goods to the customers as a part of the whole control loop. The part and the subject presented in this paper consist of the seller agents, customer agent types and the manager agent. Seller agent interacts with the customer agent according to the multi-agent approach. The interaction is based on the FIPA contract-net protocol (section 2) [FIPA, (2002)]. This simplified system was extended by disturbances influencing the agents' behavior. The disturbances occurrence is random and the number of customer agents is significantly higher than the number of seller agents. The whole system can be under these circumstances described as a stochastic system.

The behavior of agents is influenced by more randomly generated parameters (section 3.2). In this paper the influence of randomly generated parameters on the simulation outputs while using different kinds of distributions is presented. The overall workflow of the system proposed can be described as follows. The customer agents randomly generate the requests to buy some random pieces of goods. Seller agents react to these requests according to their own internal decision functions and follow the contracting. The purpose of the manager agent is to manage the requests exchange. The contracting results in the sales events to the customers. More attributes of sales like gross profit, amount, and incomes are analyzed (more can be found in section 3.3).

The results obtained show that using of inappropriate random number generation function leads to significant output data distortion. Thus, in our opinion, the selection of random generation method must be done very carefully.

This paper is structured as follows. Section 2 briefly informs about the multiagent system implementation, JADE platform and ontology. In section 3 the original simulation results are presented.

2 Multiagent System Implementation

In this section we described the setting of seller and customer agents in order to present the contracting principles. Only one stock item is being traded in the simulation and its amount is not limited – as many pieces customer wants to buy so many he gets. Sellers and customers ratio is 1:10 – one seller serves for 10 customers. Customers are in groups and each group is being served by concrete seller – their relationship is given, none of them can change his counterpart.

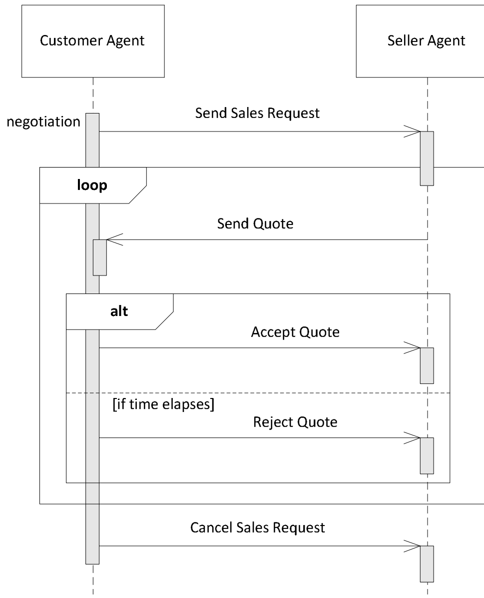


Fig. 1. Communication act between customer and seller (source: own)

Each period turn (here we assume a week) customer agent decides if he wants to buy something. His decision is defined randomly. If customer decides not to buy anything, his turn is over; otherwise he creates a sales request and sends it to his seller.

Seller agent answers with proposal message (concrete quote starting with his maximal price – limit price * 1.25). This quote can be accepted by the customer or not. Acceptation is decided from customer productivity function, which can be formalized as [Vymětal and Šperka, (2011)][1]:

$$c_n^m = \frac{\tau_n \cdot T_n \cdot \gamma \cdot \rho_m}{Z.M.y_n^{mi}} \tag{1}$$

where:

c_n^m - the price of the n-th product quoted by m-th sales representative,
τ_n - the company market share for the n-th product $0< \tau_n <1$,
T_n - the market volume for the n-th product in local currency,
γ - the competition coefficient lowering the sales success $0<\gamma \leq 1$,
ρ_m - the quality parameter of the m-th sales representative, $0{,}5 \leq \rho_m \leq 2$,
Z - the number of customers,

[1] As we think only one product to be sold and the relationship between seller and customer is 1:10 – so the customer cannot choose his seller, we have little simplify the formula.

M - the number of sales representatives in the company,
y_n^{mi} - the requested number of the *n*-th product by the *i*-th customer at *m*-th sales representative.

So the product, resp. it's price must be less or equal the customer production function. If the price is acceptable, the contract is done, otherwise no.The negotiation between the seller (sales representative of the seller – agent) is based on the contract net protocol [FIPA, (2002)].

If the price or the quantity is not accepted, rejection message is send to the seller. Seller decreases the price to the average of the limit price and current price (in every iteration is getting effectively closer and closer to the limit price) and resends the quote back to the customer. The message exchange repeats until there is an agreement or reserved time passes. Detailed schema can be seen in Fig. 1. - Communication act between customer and seller.

Seller is responsible to the manager agent. Each turn, manager gathers data from all sellers and makes state of situation of the company. This data are the result of the simulation and serves to understand company behavior in a time – depending on the agents' decisions and behavior.

Customer agents need to know some information about the market. This information is given by informative agent. This agent is also responsible for the turn management. In one word – informative agent is representing outside or controllable phenomena from the agents' perspective. In the next section the implementation platform is introduced.

2.1 JADE Platform

The agent platform JADE was chosen for implementation, because it is a real tool for rapid agent development. It contains not only communication language, but the whole platform for agents' deployment. This includes runtime environment, where agents exist, libraries to write them and also graphical tools to administrate them and monitor their state. [Wooldridge, (2009)] says that it is best-known and the most widely used.

JADE was developed by Telecom Italia in 1998 and it is still in development progress. Current version we use is 4. Used agent communication language is FIPA ACL[2]. Runtime environment running instance is called a container. At once it is possible that more than one container is running. All active containers are called a platform. Every platform has always active one special "Main container" and all other containers know where to find it. Agents are located in containers (Fig. 2. - JADE running enviroment with two containers). There are two special types of agents – AMS (Agent Management System) provides the naming service and represents the authority in the platform and the DF (Directory Facilitator) which provides Yellow Pages. Yellow Pages enable agents to seek the other ones providing the services they need to be able to achieve their goals.

[2] FIPA = the Foundation for Intelligent Physical Agents was founded 1995, in 1999 they invented ACL (Agent Communication Language).

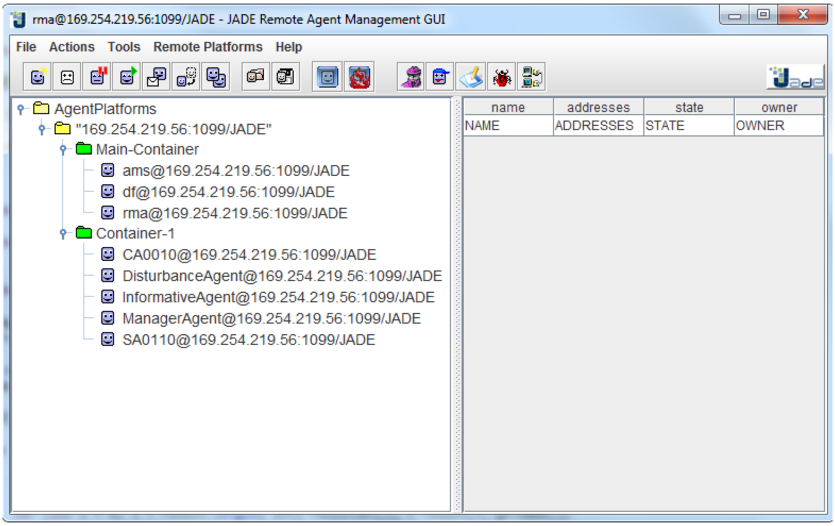


Fig. 2. JADE running enviroment with two containers (source: own)

The agent is a Java class – descendant of base JADE class Agent. Its behavior is implemented in private subclasses of the concrete Agent class extension. This behavior extends JADE class Behaviour [Bellifemine, Caire and Trucco, (2010)]. Every agent in JADE is implemented as single Java thread[3], which has influence on its encapsulation and because behaviors are private subclasses of this thread it eliminates all synchronization issues between concurrent behaviors accessing same resources and between behavior and its agent. Behavior scheduling therefore is not pre-emptive (as for Java class) but cooperative. When one behavior is in execution, it runs until returns. Programmer must define when agent switches from one behavior to another one. On the other hand the switch from one behavior to next one is very much faster than Java thread switch. In the last part of this section the implementation in details is presented.

2.2 Agents Implementation

Agents in the multiagent system are situated in two levels. Base agent, ancestor of all BPM[4] agents is the "BaseBpmAgent", which has implemented base functionality, such as registering to the yellow pages, searching for other agents, cleanup and so on. All the working agents in system are descendants of this class.

Customer agent, as in the real market, is the engine of the process. Each turn (week) it decides whether he will buy something. If so, he decides the quantity and sends the request to his seller (in JADE called CFP = call for proposal). Then above mentioned negotiation with the seller takes place and the result is the reject or the selling transaction. Once this

[3] This is very important in environments with limited resources like cellular phones.

[4] Business Process Modelling.

negotiation is done, the turn for the specific agent finishes. When all the customer agents finish their negotiations, the turn (week) is over.

To make its decision, customer agent needs information about the market[5] – here the information agent comes. This information agent is responsible for giving information about the market to customer agent (using behavior "ResponseBehavior"), but also for the timing – this agent decides when the turn (week) finishes and gives the information to all the agents to prepare for another one. Also it keeps in mind that the running model period is one year (53 weeks, using behavior "RunningTurnBehavior"). After this period it sends the info about "GameOver" and (implemented in "BaseBpmAgent") all agents finish their functionality.

The customer agent has defined 4 types of behavior:

1. **RefreshInfo** => this behavior gets the fresh information about the market and when agent is informed, it starts another, following behavior.
2. **RequestPerformer** => the negotiation with the seller agent.
3. **ReceiveMessages** => used to obtain the information from the information agent.
4. **FinishInfo** => informs the information agent that turn is over for this customer agent.

On the other hand the seller agent is sitting and waiting. Once he has a request, immediately replies to customer agent with the appropriate price and then waits again, how the communication finishes. Actually, this agent is not aware of the turn (week) itself, but via manager agent, who is asking him for the report of the week work. This agent implements 3 server behaviors:

1. **OfferRequestsServer** – reads CFP messages from the customers and negotiate with them.
2. **PurchaseOrdersServer** – sells the goods (we don't have the limitation of the goods on stock for now).
3. **InformationServer** – informs manager agent about his success in selling each week.

Once per turn manager agent asks to all the sellers how they were successful. Then values the company situation and makes the report about the state.

2.3 Ontology

During the communication between agents – they definitely need to understand each other and to be able to specify unambiguously the entity which represents the subject of their speech. "Ontology is a specification of a set of terms, intended to provide a common basis of understanding about some domain." [Wooldridge, (2009)]

---

[5] Vide customer productivity function definition – company item market share, item market volume.

JADE offers the base “Ontology” class that is extended by our implementation of singleton class “BpmOntology”. Vocabulary concepts of this ontology are using two classes – “MarketInfo” (which contains the information exchanged between the information agent and customer agent) and “SellerInfo” (which contains the information exchanged between the seller agent and customer agent). In the last section the simulation results are presented.

3 Simulations

3.1 Random and Pseudorandom Numbers

When simulating the unpredictable phenomena the multiagent systems use randomly (or pseudorandomly) generated data. They provide the critical aspect of uncertainty in a deterministic world.

The computer systems are deterministic. From its principle they are not made to do things randomly. If you want to behave them in a truly random way, some non-deterministic input must be introduced. To obtain this random data source it is possible to flip coins, pluck daisy petals or roll dice. By writing the results the database of randomness can be created. Some devices can be used to e.g. observe atmospheric noise or other unpredictable phenomenon. But it is not necessary to do such things, while these random numbers can be obtained indirectly. There are online services such as Random.org or Hotbits. But we don’t use these sources regularly for our software, because of speed. They are mostly not so fast for our uses. So we use pseudorandom numbers. They are not random at al. They are calculated by some mathematical formula.

3.2 Attributes and Different Distributions

We have chosen two important agents attributes to be generated by pseudorandom generator. These are sellers’ agent ability and customers’ agent decided quantity for purchase. We want to use different pseudorandom generators and also generate values from different types of the random distributions.

First generation will be done by using “java.util.Random” class for providing the random numbers. This Java functionality generates uniform random distribution values. If you roll dice – there is always the same probability that you get 1 or 2, 3… etc. None single outcome has higher probability than others. This distribution is appropriate to solve problems like rolling dice or shuffling cards, but most of the real world phenomena doesn’t occur with the same probability of their outcomes.

Normal distribution (Gaussian) represents the distribution where outcomes nearer the average (above or below) are more frequent than the extreme ones (extremes are not impossible, but rather unlike). Its shape is familiar bell curve. For generating this random numbers we use the Java library called Uncommon Maths written by [Dyer, (2010)]. For getting the values it uses much more random “MerseneTwisterRNG” class, which is a pure Java port of Makoto Matsumoto and Takuji Nishimura’s proven and ultra-fast Mersenne Twister Pseudo Random Number Generator for C.

3.3 Simulation Results

Goal is to simulate one year of selling / buying processes (52 turns – weeks) and to compare the results obtained from the attributes seller agent ability and customer amount of purchase from the uniform random distribution generated by basic Java class “java.util.Random” with the results obtained when generating attributes from normal distribution using Uncommon Maths. Parameters of the system were set to the following values.

On the customer side (total number of agents of this type 500):

- Maximum Discussion Turns: 10
- Mean Quantity[6]: 50
- Quantity Standard Deviation: 29

On the seller side (total number of agents of this type was 50):

- Mean Ability[6]: 0.5
- Ability Standard Deviation: 0.3
- Minimal Price: 5

On the manager side:

- Purchase Price: 4

On the market side (resp. trading company on market):

- Item Market Share: 0.5
- Item Market Volume: 5 000 000

We can see the results in the aggregated form in Table 1. There are significant differences between Pieces, Incomes, Costs and Gross Profit values in the uniform and normal table part. That means it is important to choose the right distribution of randomly generated parameters. Table 1 reflects one of the most important characteristic of the normal distribution – the values that are much closer to mean are more often in output than extremes. Standard deviations in uniform distribution results are much higher than in normal distribution generation.

Table 1. Uniform and Random distribution generation results (source: own)

Agregation	Uniform Random Distribution Generation				Normal Random Distribution Generation			
	Pieces	Incomes	Costs	Gross Profit	Pieces	Incomes	Costs	Gross Profit
SUM	4382,00	26975,00	17528,00	9447,00	1969,00	12306,25	7876,00	4430,25
AVG	84,27	518,75	337,08	181,68	37,87	236,66	151,46	85,20
Std.dev.	35,42	215,46	141,70	74,24	14,68	91,73	58,70	33,02

[6] Mean and standard deviation were decided according previous many simulations based on uniform distribution.

Because the uniform data are more spread, the final values are higher, but for the more realistic scope of the generation both attributes – seller ability and also the selected quantity should be taken from the normal distribution rather than from uniform. Most of the sellers will be average – near the mean, extremes are rare (even if in positive meaning of word can be a real treasure for the company).

We took the main significant data series – Incomes and made the correlation analysis on both results. The correlation coefficient -0.26 doesn't prove tight correlation binding between the results of generation with different random attributes distribution.

4 Conclusion

The results achieved show that using inappropriate random number generation function leads to significant output data distortion, so the generation function selection must be done very carefully. Comparison was done on the uniform versus normal distribution of data. The uniform distribution is usable for values with the same probability of output – such as rolling a dice, flipping a coin or shuffling a deck of cards. On the other hand – the normal distribution can be used for many phenomena which occur in normal world – ability of sellers, productivity of workers, competitiveness of companies etc. All these are real candidates to be simulated with data generated according to the normal distribution, because most of the occurrences will be near the mean – not uniformly distributed in a whole range of outputs. Different distributions can be used for different purposes.

The implementation of the whole company's control loop will be the task for the future research.

References

[1] Bellifemine, F., Caire, G., Trucco, T.: Jade Programmer's Guide. Java Agent Development Framework (2010), `http://jade.tilab.com/doc/programmersguide.pdf`

[2] De Snoo, C.: Modelling planning processes with TALMOD. Master's thesis, University of Groningen (2005)

[3] Dyer, D.W.: Uncommons Maths - Random number generators, probability distributions, combinatorics and statistics for Java (2010), `http://maths.uncommons.org/`

[4] Foundation for Intelligent Physical Agents (FIPA), FIPA Contract Net Interaction Protocol. In Specification (online). FIPA (2002), `http://www.fipa.org/specs/fipa00029/SC00029H.pdf` (cit. June 13, 2011)

[5] Jennings, N.R., Faratin, P., Norman, T.J., O'Brien, P., Odgers, B.: Autonomous agents for business process management. Int. Journal of Applied Artificial Intelligence 14, 145–189 (2000)

[6] Macal, C.M., North, J.N.: Tutorial on Agent-based Modeling and Simulation. In: Proceedings: 2005 Winter Simulation Conference (2005)

[7] Moreno, A., Valls, A., Marín, M.: Multi-agent Simulation of Work Teams. In: Mařík, V., Müller, J.P., Pěchouček, M. (eds.) CEEMAS 2003. LNCS (LNAI), vol. 2691, pp. 281–291. Springer, Heidelberg (2003)
[8] Scheer, A.-W., Nüttgens, M.: ARIS Architecture and Reference Models for Business Process Management. In: van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 376–389. Springer, Heidelberg (2000)
[9] Sierhuis, M.: Modeling and Simulating Work Practice. PhD thesis, University of Amsterdam (2001)
[10] Vymetal, D., Sperka, R.: Agent-based Simulation in Decision Support Systems. In: Proceedings of Distance Learning, Simulation and Communication (2011) ISBN 978-80-7231-695-3
[11] Wolf, P.: Úspěšný podnik na globálním trhu. CS Profi-Public, Bratislava (2006)
[12] Wooldridge, M.: MultiAgent Systems: An Introduction to, 2nd edn. John Wiley & Sons Ltd., Chichester (2009)
[13] Yan, Y., Maamar, Z., Shen, W.: Integration of Workflow and Agent Technology for Business Process Management (2001)

A Multi-context Representation of Mental States

Aldo Franco Dragoni

Università Politecnica delle Marche

Abstract. We provide a simple computational structure for representing recursive *mental states* that could constitute the semantic level of a formal language to deal with the dynamics of mental states. For the sake of simplicity, we base our conception of mental state on only two primitive attitudes: *beliefs* (for the cognitive sphere) and *desires* (for the volitional sphere).

1 Introduction

With “Belief Desires Intentions” architectures (BDI), the nature of mental states of agents is obviously recursive, since an agent has to represent the mental state of the others which, in turn, have to represent its mental state and so on. For instance, Adam needs to know that Eve believes that he wants to stay in the Eden, etc. As an alternative to the classical modal approach to BDI architectures (as the one introduced in [1]) we propose a simple computational structure for representing recursive mental states. This computational structure improves the clausal-based formalism introduced in [5] and constitutes the semantic level of a representational language that will make it easier to deal with the dynamics of mental states (i.e. with the theory of Speech Acts).

For the sake of simplicity, we base here our conception of mental state on only two primitive attitudes: *beliefs* (for the cognitive sphere) and *desires* (for the volitional sphere). "Believing" just means "having a symbolic representation of a real state of the world", while "desiring" means "having a symbolic representation of an ideal state of the world", where "ideal" means "preferable over all the possible states of the world". Other attitudes, as “knowledge” and “intention”, might either be added or eventually derived from these primitives (see the discussion in [2]).

Let us introduce the mental structure of our artificial agent Adam. It believes a certain state of affairs B of the world while it desires a state D. Both B and D might be partitioned into an external (E) and an internal (I) continents, the latter referring to the agent’s own mental world. Thus we have four continents, BE, BI, DE and DI. Normally BE≠DE; in fact, BE=DE means that Adam is completely satisfied with the external world where he believes to live in. Analogously, BI and DI might be different, meaning that Adam desires to change what he believes its own mental state is (Fig 1). Both the continents BI and DI may be recursively mapped into four subcontinents, two of which are meant to be “known” (internal and external) and the other two are intended to be “desired” (internal and external). Clearly, this notion of mental state is recursive: mental attitudes may have other mental attitudes as arguments and so on (Fig. 2). Although recursion is theoretically infinite, three or four levels of depth should be sufficient to describe most of the psychologically plausible human mental dynamics (see also the discussion in [3]).

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 350–359, 2012.

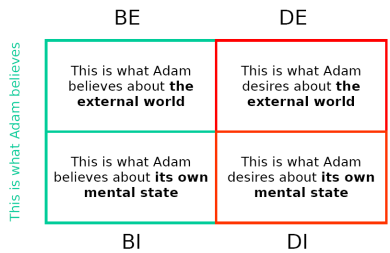


Fig. 1. Partitioning mental *worlds* into ***internal*** and ***external*** *continents*

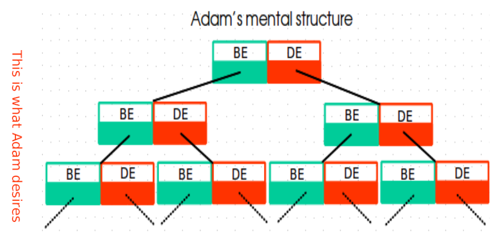


Fig. 2. Internal continents are nested recursively and the recursion is potentially infinite

When Adam is engaged in a conversation with Eve, both its internal continents take into account also Eve's mental structure. On her side, Eve's internal continents must map also Adam's mental structure and so on (Fig 3). Adam&Eve's scenario might be generalized to a group of agents. Let us denote with $\Sigma_{i,G}$ the mental structure of the agent i as part of the group G. Since mental structures are trees, pathnames are the obvious way to identify nodes.

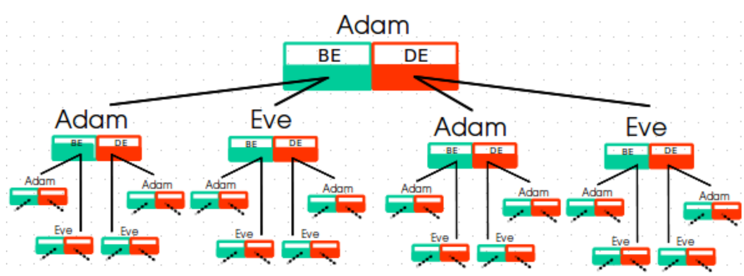


Fig. 3. Adam's mental state after the creation of Eve

2 The Multi-contextual Representation of a Mental State

Multi-language formalisms [4] gave us new opportunities to represent mental states. We base the semantics on the notion of mental structure given in the previous section. The formalism is modular; its elementary brick is a triangular structure, named x-module, made of three contexts (i.e. three different sets of formulae, Fig 4 right side):

- x: represents the agent x's mental state as a whole (the *mental context*)
- B: represents x's beliefs (the *belief context*)
- D: represents x's desires (the *desire context*)

Sentences in B and in D represent, respectively, something which is "consciously believed" or "consciously desired". Instead, non-atomic formulae in x (e.g., implications between beliefs and/or desires) should represent mental attitudes and correlations that affect unconsciously the agent's cognitive behaviour, i.e. mental conditions that the agent is not aware of. We call the belief context and the desire context the *argumental contexts* of the *x*-module, for reasons that will be cleared later. Each context has to be a decidable theory (possibly empty) expressed in an appropriate formal language (that will be discussed in the next section). A fundamental component of any multi-context formalism is the "*bridge-rule*". Each of the four arrows in the x-module represents a bridge-rule, namely *B_down*, *B_up*, *D_down* and *D_up*. Informally, their effect is as follows:

- if it is provable in x that a formula is *believed*, then that formula is provable in B (*B_down*) and vice-versa (*B_up*)
- if it is provable in x that a formula is *desired*, then that formula is provable in D (*D_down*) and vice-versa (*D_up*)

These bridge rules' semantics will be formally described in the next section. The *x*-module is then replicated and assembled in a hierarchical manner with the dummy bridge rule illustrated in. Fig. 5 (which shows the multi-contex representation of Adam's mental state when engaged in a communication with Eve, again truncated at the third level of depth).

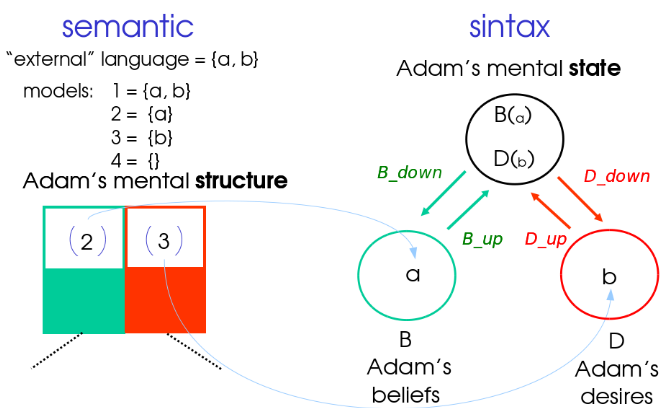


Fig. 4. Our multi-context formalism for representing mental states and its semantics

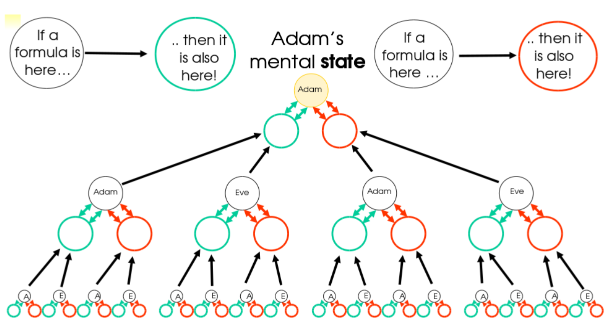


Fig. 5. The triangular module is the elementary brick of an infinite recursive syntactic structure

Let us denote with $MS_{i,G}$ the multi-context representation of the agent *i*'s mental state, being *i* part of the group *G*. The correspondence between $MS_{i,G}$ and $\Sigma_{i,G}$ is quite evident, and pathnames are again the obvious way to label the contexts. Just to give an intuitive idea of the meaning of the various contexts:

- *a*/D/*a* represents what *a* desires its own mental state would be
  - (in particular) *a*/D/*a*/B represents what *a* desires to believe
  - (in particular) *a*/D/*a*/D represents what *a* desires to desire
- *a*/D/*e* represents what *a* desires *e*'s mental state would be
  - (in particular) *a*/D/*e*/B represents what *a* desires that *e* believes
  - (in particular) *a*/D/*e*/D represents what *a* desires that *e* desires
- *a*/B/*a* represents what *a* believes its own mental state is
  - (in particular) *a*/B/*a*/B represents what *a* believes to believe
  - (in particular) *a*/B/*a*/D represents what *a* believes to desire
- *a*/B/*e* represents what *a* believes *e*'s mental state is
  - (in particular) *a*/B/*e*/B represents what *a* believes that *e* believes
  - (in particular) *a*/B/*e*/D represents what *a* believes that *e* desires

3 Languages and Semantics

Let us define now the languages adopted in the contexts and their semantics.

3.1 Syntax

To cope with the complexity of this formalism we need three kinds of languages. At the basic level there are the *external languages*, i.e. those which agents adopt to represent the external world. Each agent *i* may speak its own external language *ELi*. Although the essential requirement for each *ELi* is decidability, for practical purposes we suppose these languages to be propositional. Referring to the mental structure depicted in Fig 4, *ELa* is used in all the continents **a*/BE and * *a*/DE (where * stands for "any path") and *ELe* is "spoken" in all the continents **e*/BE and **e*/DE.

Definition 1. For each agent *i*, the ***external language*** *ELi* is the propositional language used by *i* to represent objects and relations of the world around it.

External languages are the basis for the *argument languages* adopted in the contexts **i*/B and **i*/D.

Definition 2. For each agent $i \in G$, its ***argument language*** *ALi* is defined as follows:

1. α is an *atomic* formula of *ALi* iff it satisfies of the following two:
   (a) $\alpha \in ELi$
   (b) $\alpha \in MLj$ where *MLj* is the *mental language* (defined below) of an agent $j \in G$
2. $\neg\alpha$ is a formula of *ALi* iff α is a formula of *ALi*
3. $\beta\vee\alpha$, $\beta\wedge\alpha$ and $\beta\rightarrow\alpha$ are formulae of *ALi* iff α and β are formulae of *ALi*
4. nothing else is a formula of *ALi*.

Definition 3. For each agent *i*, its ***mental language*** *MLi* is defined as follows:

1. $iB(_\delta)$ and $iD(_\delta)$ are *atomic* sentences of *MLi* iff δ is a sentence of *ALi*
2. ¬α is a formula of *MLi* iff α is a formula of *MLi*
3. β∨α, β∧α and β→α are formulae of *MLi* iff α and β are formulae of *MLi*
4. nothing else is a formula of *MLi*.

Although $iB(_\delta)$ and $iD(_\delta)$ are *atomic*, we distinguish between the "mental level", *iB*( ) or *iD*( ), and the "argument-level" $_\delta$, where the subscript δ is an individual constant of *MLi* denoting the name of a formula δ∈*ALi*. These reified formulae are the only constants of *MLi*, and there are no variables. The interleaving between *MLi* and *ALi* is to avoid that sentences of the form $iB(_\delta)$ (or $iD(_\delta)$), with δ∈*ELj* and *i*≠*j* could belong to *MLi* and appear in the contexts */*i*; thus it is not allowed that an agent *i*'s mental attitude has as argument a state of the external world described through the language of another agent *j*. Even if all the agent speak the same external language, mental languages would be completely disjoint from each others. In other words, there does not exist a formula belonging to more than one mental language. This is just because each *MLi* relies on its own set of atomic sentences built upon its own couple of unary predicates *iB*( ) and *iD*( ). The argument languages of the various agents differ only in the part relative to their external language. For any agent *i*, any formula of *MLi* is also a formula of *ALi*.

3.2 Semantics

As anticipated, theories in contexts **i* are expressed through *MLi*, while theories in contexts **i*/B and **i*/D are expressed through *ALi*. The fact that a formula α belongs to the theory contained in a context *C* will be expressed by prefixing the name of that context as a label to that formula, i.e. *C*:α. These *labelled formulae* are of the following kinds:

- **i*:α where α∈*MLi* (*mental formulae*)
- **i*/B:α where α∈*ALi* (*argument formulae*)
- **i*/D:α where α∈*ALi* (*argument formulae*)

To simplify definitions and proofs we'll refer to a generic predicate *iP*( ), instead of *iB*( ) or *iD*( ), and to a generic context P, instead of B or D. Let us start by defining the semantics of *atomic* mental formulae, i.e., the semantics of $*i{:}iP(_\delta)$ where δ∈*ALi*.

Definition 4. (Bridge Rules Up and Down) An *atomic* mental formula $*i{:}iP(_\delta)$ of *MLi*, with δ∈*ALi*, is true iff the argument formula **i*/P:δ is true.

Definition 5. Given two mental formulae α and β of *MLi*:

1. **i*: ¬α is true iff **i*:α is false.
2. **i*: α∨β is true iff either **i*:α is true, or **i*:β, or both
3. **i*: α∧β is true iff both **i*:α and **i*:β are true
4. **i*: α→β is true iff **i*: ¬α∨β is true.

Definition 4 grounds the semantics of mental formulae (*MLi*) upon the semantics of argument formulae (*Ali*, same agent *i*), while definition 2 defines the semantic of argument languages *ALi*, in turn, upon that of atoms of *ELi* and *MLj* (eventually of *another* agent $j \neq i$). Hence the semantic of this multi-language formalism stands essentially on the atoms $*i$/P:α with $\alpha \in ELi$, which are related to the notion of "mental structure" introduced in the previous section. This mental structure maps on the multi-context representations of mental states since contexts of the type $*i$/B maps into continents of the type $*i$/BE, while context of the type $*i$/D maps into continents of the type $*i$/DE. Let PE denote either BE or DE. Let Ω_i denote the set of all the "truth functions" over the propositional letters of *ELi*, i.e. the set of all the possible *interpretations* of *ELi*.

Definition 6. A ***model*** of a continent $*i$/PE in $\Sigma_{i,G}$ is a subset of Ω_i.

By default continents *I/PE in $\Sigma_{i,G}$ have the overall Ω_i as a model, so that nothing can be derived in their corresponding contexts $*i$/P in $MS_{i,G}$, apart from tautologies. By doing so our agents will have a void mental state at birth! If as model of a continent is void, then the corresponding context is contradictory. A mental structure is a theoretically infinite tree, so we need to provide semantics for any node of that tree.

Definition 7. A Truth Tree $TT_{i,G}$ for an agent *i* in a group *G*, is the assignment of a model to each continent $*i$/PE in $\Sigma_{i,G}$.

In the rest of the paper we always refer to an agent *i* as member of a group *G*, so we undertake both the agent and the group by writing *TT*, MS and Σ instead of, respectively, $TT_{i,G}$, $MS_{i,G}$ and $\Sigma_{i,G}$. Let Ω (without the index) denote the set of all the Truth Trees *TT*. The semantics of argument atoms $*i$/P:α, where $\alpha \in ALi$ is a formula of *ELi*, is defined as follows:

Definition 8. Let $*i/\mathrm{P}_{TT}$ denote the model associated by the truth tree *TT* to the continent $*i$/PE. We say that $TT \in \Omega$ is a model for an *argument* formula $*i$/P:α, where $\alpha \in ALi$ is a formula of *ELi*, iff α is true in $*i/\mathrm{P}_{TT}$, i.e. iff α is true for *all* the interpretations in the set $*i/\mathrm{P}_{TT}$. In this case we write $TT \Vdash *i/\mathrm{P}:\alpha$ and $*i/\mathrm{P}_{TT} \Vdash \alpha$.

Note that, while $*i$/P:α and $*i$/P:$\neg\alpha$ cannot both be true (unless $*i/\mathrm{P}_{TT}$ is void), they may both be false, since $*i/\mathrm{P}_{TT}$ could be a model neither for $*i$/P:α nor for $*i$/P:$\neg\alpha$. This case provides apparently another semantics for the *negative* mental attitude, i.e.:

Definition 9. $TT \in \Omega$ is a model for a *mental* formula $*i{:}\neg iP(\delta)$, where $\delta \in ALi$ is a formula of *ELi*, iff *it is not the case that* δ is true in *all* the interpretations of the set $*i/\mathrm{P}_{TT}$. In this case we write $TT \Vdash *i{:}\ \neg iP(\delta)$, $*i/\mathrm{P}_{TT} \nVdash \delta$ and $TT \nVdash *i/\mathrm{P}:\delta$.

If δ is true *only in some* interpretations of $*i/\mathrm{P}_{TT}$, then $\neg\delta$ is true in the other interpretations, so that neither $*i{:}iP(\delta)$ nor $*i{:}iP(\neg\delta)$ are true, hence from definition 5 $*i{:}\neg iP(\delta)$ and $*i{:}\neg iP(\neg\delta)$ are both true! This would informally mean that the agent has no evidence (or no desire) about δ. However this result can also be reached from the other way around since, if neither $*i$/P:δ nor $*i$/P:$\neg\delta$ are true, then from definition 4 neither $*i{:}iP(\delta)$ nor $*i{:}iP(\neg\delta)$ are true.

Let's now go to the case of argument atoms *i/P:α, where $\alpha \in ALi$ is a mental formula of *MLj*, for any agent *j*; their semantics is very simple:

Definition 10. An attitudinal formula *i/P:α, where $\alpha \in ALi$ is a formula of some mental language *MLj*, is true iff α is true in the context *i/P/j.

This definition makes recursive the semantic of mental formulae of *MLi* in *i. In fact, it depends on the semantics of argument formulae of *ALi* in *i/P, which in turn may depend on the semantics of mental formulae of *MLi* in *i/P/i and so on. The semantics of non-atomic argument formulae *i/P:α, where $\alpha \in ALi$, is as usual:

Definition 11. Given two formulae α and β of *ALi*:

1. *i/P: $\neg\alpha$ is true iff *i/P:α is false.
2. *i/P: $\alpha \vee \beta$ is true iff either *i/P:α is true, or *i/P:β, or both
3. *i/P: $\alpha \wedge \beta$ is true iff both *i/P:α and *i/P:β are true
4. *i/P: $\alpha \rightarrow \beta$ is true iff *i/P: $\neg\alpha \vee \beta$ is true.

4 Uncertainty and Undeterminism. Logical Consequence

"Believing $\alpha \wedge \beta$" is (psyco)logically equivalent to "Believing α $\wedge$ Believing β", but "Believing $\alpha \vee \beta$" is not (psyco)logically equivalent to "Believing α $\vee$ Believing β"! For instance, "Believing that President Obama will withdraw from Afghanistan or he will not" is not (psyco)logically equivalent to "Believing that Obama will withdraw from Afghanistan $\vee$ Believing that Obama will not withdraw from Afghanistan". In fact, the former mental attitude is simply "believing a tautology" (hence believing a certainty), while the latter is, more puzzingly, a disjunction of two different beliefs, i.e. an *undetermined mental attitude*! The same consideration applies to the "Desiring" mental attitude. So, we want a formal system for mental states in which undeterminism is different from uncertainty, i.e.:

$$*i{:}(iP(_{\alpha}) \vee iP(_{\beta})) \neq *i{:}iP(_{\alpha \vee \beta})$$

and this multicontext formalism satisfies this requirement:

Theorem 1. For any context *i, the equivalence $*i{:}(iP(_{\alpha}) \vee iP(_{\beta})) \neq *i{:}iP(_{\alpha \vee \beta})$ does not hold. It holds instead the material implication

$$*i{:}(iP(_{\alpha}) \vee iP(_{\beta})) \rightarrow *i{:}iP(_{\alpha \vee \beta})$$

Proof. Let us show first that $*i{:}(iP(_{\alpha}) \vee iP(_{\beta})) \rightarrow *i{:}iP(_{\alpha \vee \beta})$.

1. $*i{:}iP(_{\alpha})$ implies *i/P:α (definition 4)
2. *i/P:α implies *i/P:$\alpha \vee \beta$ (definition 9 and semantic of the logical connective $\vee$)
3. *i/P:$\alpha \vee \beta$ implies $*i{:}\ iP(_{\alpha \vee \beta})$ (definition 4)
4. $*i{:}iP(_{\alpha})$ implies $*i{:}\ iP(_{\alpha \vee \beta})$ (1, 2 and 3)
5. $*i{:}iP(_{\beta})$ implies $*i{:}\ iP(_{\alpha \vee \beta})$ (4 with β in the place of α)
6. $*i{:}(iP(_{\alpha}) \vee iP(_{\beta}))$ implies $*i{:}\ iP(_{\alpha \vee \beta})$ (4, 5 and definition 5 point 2)

But the opposite $*i{:}iP(_{\alpha \vee \beta}) \rightarrow *i{:}(iP(_{\alpha}) \vee iP(_{\beta}))$ does not hold. As a counterproof consider $*i/P_{TT} \equiv \{2,3\}$ in the example of Fig. 10; with such a model $*i{:}\ iP(_{\alpha \vee \beta})$ is true but neither $*i{:}iP(_{\alpha})$ nor $*i{:}iP(_{\beta})$ are verified.

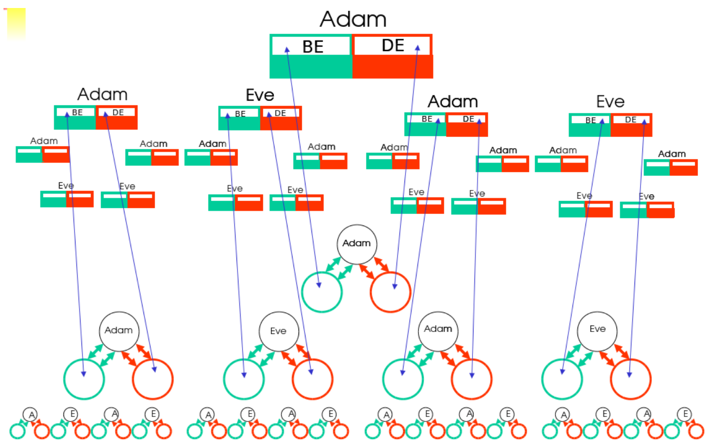


Fig. 6. Our multi-context formalism bases its semantic on the corresponding mental structure

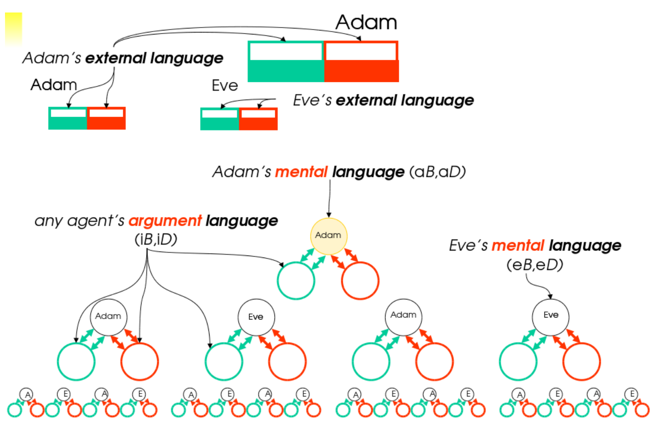


Fig. 7. External language, *Argument* language and *Mental* language

However, our semantics does not give support for undeterminism! We can prove logical properties about formulae of the scheme $*i{:}(iP(_{\alpha}) \vee iP(_{\beta}))$, but we cannot *produce* them with our semantics! We can derive $*i{:}(iP(_{\alpha}) \vee iP(_{\beta}))$ from $*i{:}iP(_{\alpha})$ or $*i{:}iP(_{\beta})$, but we cannot have the former mental attitude without (at least) one of the latters. This means that our mental states are completely determined! However, in a forthcoming (hopefully) paper we'll show how it is possible to introduce undeterminism by extending our semantics with the introduction of *constraints* over the possible Truth Trees. Notwithstanding this problem (that we cannot originate mental disjunctions but only logically derive them from their disjuncts), this formalism satisfies classical properties of logical systems.

Theorem 2. For any context *, if $*:\alpha$, $*:\beta$ and $*:\chi$ are syntactically correct labelled formulae, then the following classical equivalence rules are preserved:

- $*:\neg(\alpha\wedge\beta) \equiv *:\neg\alpha\vee\neg\beta$ (De Morgan)
- $*:\neg(\alpha\vee\beta) \equiv *:\neg\alpha\wedge\neg\beta$ (De Morgan)
- $*:\alpha\wedge(\beta\vee\chi) \equiv *:(\alpha\wedge\beta)\vee(\alpha\wedge\chi)$ (Distributive of $\wedge$ over $\vee$)
- $*:\alpha\vee(\beta\wedge\chi) \equiv *:(\alpha\vee\beta)\wedge(\alpha\vee\chi)$ (Distributive of $\vee$ over $\wedge$)

Proof. (see [5]).
With these definitions, the semantics of any labelled formula $*:\alpha$ is completely grounded on the notion of Truth-Tree. As usual, the semantic of $\wedge$ is extended to the semantics of sets of labelled formulae.

Definition 12. A Truth Tree *TT* is a model for a set of labelled formulae *S*, written as $TT \Vdash S$, iff *TT* is a model for any element of *S*.

The fundamental notion of logical consequence is trivially extended to labelled formulae.

Definition 13. Given a set *S* of labelled formulae, let [*S*] denote the set of its models. $*:\alpha$ is defined a logical consequence of *S*, written $S\models *:\alpha$, iff $[S] \subseteq [\{*:\alpha\}]$.

The previous definition means that all the models of *S* are also models of the (singleton) set $\{*:\alpha\}$. In the following, when *S* is singleton, i.e. $S=\{*:\alpha\}$, we will write $[*:\alpha]$ instead of $[\{*:\alpha\}]$. Let us notice the difference between the two symbols:

Definition 14. Two labelled formulae $*:\alpha$ and $*:\beta$ are said to be *logically equivalent*, written $*:\alpha \equiv *:\beta$, when $*:\alpha\models *:\beta$ and $*:\beta\models *:\alpha$, i.e., iff $[*:\alpha]=[*:\beta]$. Analogously, two sets of labelled formulae *S* and *R* are logically equivalent iff $[S]=[R]$. A labelled formula $*:\alpha$ is said to be *valid* when $[*:\alpha]=\Omega$, i.e. iff $\forall_{TT}\ TT\Vdash *:\alpha$.

Definition 15. Any mental attitude of the agent *i* can be represented as a labelled formula $*:\alpha$, where $\alpha\in ALi$, and vice-versa.

As long as we represent mental attitudes with labelled formulae, we can represent mental states with sets of labelled formulae. From definition 12 we see that a Truth Tree is *TT* is a model for a mental state, iff *TT* is a model for any of its mental attitudes. This definition sounds like the semantics of the logical connective $\wedge$. One may be tempted to conclude that "a mental state is a logical conjunction of mental attitudes". Unfortunately, we didn't define a language to which a pseudo-formula like $C_1:\alpha_1 \wedge C_2\alpha:_2 \wedge \ldots \wedge C_n:\alpha_n$ could belong to, but we can define a particular conjunctive labelled formula to which a mental state is logically equivalent.

Theorem 3. For any set of labelled formulae $\{C_1:\alpha_1,\ldots,C_n:\alpha_n\}$ of the multi-context representation MS of the agent *i*'s mental state, there exists a labelled formula $i:\beta_1\wedge\ldots\wedge\beta_n$ such that $\{ C_1:\alpha_1,\ldots,C_n:\alpha_n\} \equiv i:\beta_1\wedge\ldots\wedge\beta_n$.

Proof. (see [5])
Hence, any agent *i*'s mental state can be completely represented by a single labelled formula $i:\alpha$, where $\alpha\in MLi$. Now that we have a well-grounded semantics, we need an

efficient syntactic theorem-prover, which has to be sound and complete. "Sound" means that if it proves a labelled formula to be valid, that must actually be true semantically or, which is the same, that every time it proves that a labelled formula derives from others that formula must be a logical consequence of the latters. "Complete" means that for every valid labelled formula there should be somewhere a derivation of that formula from a void set of labelled formulae (i.e. from a void mental state, which means also from any mental state) or, which is the same, that every time a labelled formula is a logical consequences of some others the theorem prover is able to prove that in some way. Formally:

Definition 16. A theorem-prover $\vdash$ for labelled formulae is sound and complete iff for any $*{:}\alpha$:

$\vdash *{:}\alpha$ iff $\forall_{TT}\, TT \Vdash *{:}\alpha$

or, which is the same:

$C_1{:}\alpha_1 \vdash C_2{:}\alpha_2$ iff $C_1{:}\alpha_1 \vDash C_2{:}\alpha_2$

5 Future Work

1. We need to introduce *constraints* over Truth Trees. They will be necessary to provide a semantic for logical relations between different mental attitudes. For instance, if we want to model a mental rule such that "if the captain of the vessel *believes* that the weather is getting worse and he *believes* that the sea will reach the grade 6 of the Douglas scale then he *desires* to remain in port", then we have to impose constraints over the possible Truth Trees.
2. We need to build a correct and complete theorem prover.
3. We need to justify the effort spent in building this formalism by showing how it favours the Dynamics of Mental States, i.e. how it provides a semantic to the production of a "speech act" inside the speaker's mental state, and to the interpretation of a "speech act" in the listener's mental state (as done in [6]).

References

1. Cohen, P.R., Levesque, H.J.: Intention is choice with commitment. Artificial Intelligence 42, 213–261 (1990)
2. Rao, A.S., Georgeff, M.: BDI Agents:from theory to practice. In: Proceedings the First International Conference on Multi-Agent Systems (ICMAS 1995), S. Francisco, CA, pp. 312–319 (1995)
3. Gmytrasiewicz, P.J., Durfee, E.H.: A Rigorous, Operational Formalization of Recursive Modeling. In: Proceedings of the First International Conference on Multi-Agent Systems (ICMAS 1995), S. Francisco, CA, pp. 125–132 (1995)
4. Giunchiglia, F., Serafini, L.: Multilanguage Hierarchical Logics (or:How we can do without modal logic). Artificial Intelligence 65, 29–70 (1994)
5. Dragoni, A.F.: Mental states as multi-context systems. Annals of Mathematics and Artificial Intelligence 54(4), 265–292 (2009)
6. Dragoni, A.F., Giorgini, P., Serafini, L.: Mental States Recognition from Communication. Journal of Logic and Computation 12(1), 119–136 (2002), doi:10.1093/logcom/12.1.119

Epistemic Profiles and Belief Structures*

Barbara Dunin-Kęplicz[1,2] and Andrzej Szałas[1,3]

[1] Institute of Informatics, University of Warsaw, Poland
{keplicz,andrzej.szalas}@mimuw.edu.pl
[2] Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland
[3] Department of Computer and Information Science, Linköping University, Sweden

Abstract. The paper is devoted to a novel formalization of beliefs in multiagent systems. Our aim is to bridge the gap between idealized logical approaches to modeling beliefs and their actual implementations. Therefore the stages of belief acquisition, intermediate reasoning and final belief formation are isolated and analyzed. We give a novel semantics reflecting those stages and suitable for building complex belief structures in the context of incomplete and/or inconsistent information. Namely, an agent starts with constituents, i.e., sets of initial beliefs acquired by perception, expert supplied knowledge, communication with other agents and perhaps other ways. Next, the constituents are transformed into consequents according to agents' epistemic profiles.

Additionally, a uniform treatment of single agent and group beliefs is achieved. Importantly, we indicate an implementation framework ensuring tractability of reasoning about beliefs.

1 Beliefs in Multiagent Systems

In multiagent systems, agents' awareness is typically expressed in terms of beliefs built on various forms of observations, communication and reasoning [1,7,8,9,19]. Existing logic-based approaches [10,12,13,14] usually lead to high complexity of reasoning, unacceptable from the point of view of their implementation and use. In fact, the underlying semantical structures are rather abstract and do not really reflect the way beliefs are acquired and finally formed. Also, in many applications one needs to take into account relevant features of perception, including:

- limited accuracy of sensors and other devices;
- restrictions on time and other resources affecting measurements;
- unfortunate combinations and unpredictability of environmental conditions;
- noise, limited reliability and failure of physical devices.

During belief formation initial and intermediate beliefs are confronted with other beliefs originating from a variety of sources. The final beliefs can then substantially deviate from the initial ones. When belief formation is completed there might still exist areas of agents' ignorance and inconsistencies. Despite such partially unsettled information, agents need to make individual or collective decisions and act accordingly. Thus, reducing the areas of ignorance and inconsistencies is vital. This can be accomplished in

* Supported by the Polish National Science Centre grants 2011/01/B/ST6/02769 and 6505/B/T02/2011/40.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 360–369, 2012.

many ways, including a variety of reasoning methods, belief exchange by communication, belief fusion or supplementary observations.

Of course, there is no guarantee to acquire the whole necessary information or to resolve all inconsistencies. Information may still remain partly unknown or inconsistent. In order to adequately deal with such situations, two issues need to be addressed:

- paraconsistent models allowing for inconsistencies and lack of information;
- nonmonotonic reasoning techniques for completing missing information and resolving inconsistencies.

However, both paraconsistent and nonmonotonic reasoning [3,11,15,18] remain intractable. This naturally restricts their use in multiagent systems and calls for a shift in perspective. We propose a novel framework for flexible modeling beliefs of heterogenous agents, inspired by knowledge representation and deductive database techniques.

In order to ensure flexibility, individual characteristics of agents are reflected in the diversity of *epistemic profiles* characterizing both their reasoning capabilities as well as the manner of dealing with conflicting or lacking information. Naturally, agents reach conclusions by combining various forms of reasoning, including belief fusion, disambiguation of conflicting beliefs or completion of lacking information. This rich repertoire of available methods enables for heterogeneity of agents' reasoning characteristics.

The rest of this paper is structured as follows. In Sections 2 and 3 our approach to structuring beliefs is outlined and motivated. Next, in Section 4, formal syntax and semantics of the basic language are defined. The pragmatics of its use in multiagent systems is discussed in Section 5. Finally, Section 6 concludes the paper.

2 Structuring Beliefs

In the sequel, belief formation by agents will be unveiled. In the idealized logical approaches to agency, this paradigmatic part of agents' activity seems to be, at least partly, neglected. We analyze and model this process from the very beginning, that is from agents' perception and other kinds of basic beliefs.

The basis for the framework is formed by semantical structures reflecting the processes of agent's belief acquisition and formation. Namely, an agent starts with *constituents*, i.e., sets of beliefs acquired by perception, expert supplied knowledge, communication with other agents and perhaps other ways. Next, the constituents are transformed into *consequents* according to agent's *individual epistemic profile*

While building a multiagent system, lifting beliefs to the group or even more complex level is substantial. As regards belief formation, we intend to reach a conceptual compatibility between individual and group cases. Assuming that *the group epistemic profile* is set up, analogical individual and group procedures are then applicable for defining belief fusion methods, where:

- consequents of group members become constituents at the group level;
- such constituents are further transformed into group consequents.

Observe that this way various perspectives of agents involved are taken into consideration and merged. Moreover, the underlying semantical structures for groups and individuals. In fact, the same uniform approach applies to groups of groups of agents or to mixed groups of individuals and other complex topologies. The only requirement is that all epistemic profiles of complex structures are fixed.

3 An Example

Consider an agent equipped with a sensor platform for detecting air pollution and two different sensors for measuring the noise level. The agent has also some information about the environment, including places in the neighborhood, etc. The task is to decide whether conditions in the tested position are healthy.

It is natural to consider, among others, three constituents:

- C_p gathering beliefs about air pollution at given places, in terms of $P(x, y)$ indicating the pollution level y at place x, where $y \in \{\mathit{low}, \mathit{moderate}, \mathit{high}\}$;
- C_n gathering beliefs about noise level at given places, in terms of $N_i(x, y)$ indicating the noise level y at place x, as measured by a sensor $i \in \{1, 2\}$, where $y \in \{\mathit{low}, \mathit{moderate}, \mathit{high}\}$;
- C_e gathering information about the environment in terms of $Cl(x, y)$ indicating that place x is close to a place characterized by y, where $y \in \{\mathit{pollutive}, \mathit{noisy}, \mathit{neutral}\}$.

For example, we may have:

$$C_p = \{P(a, \mathit{low})\}, \quad C_n = \{N_1(a, \mathit{high})\},$$
$$C_e = \{\neg Cl(a, \mathit{noisy}), Cl(a, \mathit{neutral}), Cl(a, \mathit{pollutive})\}.$$

Note that we have no information from the second noise sensor (no literal $N_2()$ is given) and somehow inconsistent information as to the pollution level (C_p indicates low level, but according to C_e the agent is close to a pollutive place).

Based on constituents, the agent has to decide whether the situation is healthy or not (and include thus obtained belief to the set of consequents). For example, the agent may accept $F = \{\neg S(a, \mathit{healthy}), S(a, \mathit{healthy})\}$ as its consequent, i.e., it may have inconsistent beliefs about the issue whether the situation at place a is healthy.

4 Syntax and Semantics

To model real-world phenomena, a commonly used logic is the four-valued logic proposed in [2]. However, as discussed, e.g., in [5,20], this approach is problematic as it often provides results deviating from intuitions. Our approach is strongly influenced by ideas underlying the 4QL query language [16,17] which does not share such problems. 4QL enjoys tractable query computation and captures all tractable queries. Moreover, it provides simple, yet powerful constructs for expressing nonmonotonic rules reflecting, among others, default reasoning, autoepistemic reasoning, defeasible reasoning, local closed world assumption, etc. [16]. Therefore, 4QL is a natural implementation tool creating a space for a diversity of applications.[1]

In what follows all sets are finite except for sets of formulas.

We deal with the classical first-order language over a given vocabulary without function symbols. We assume that *Const* is a fixed set of constants, Var is a fixed set of variables and Rel is a fixed set of relation symbols.

[1] An interpreter of 4QL is available via 4ql.org.

A *literal* is an expression of the form $R(\bar{\tau})$ or $\neg R(\bar{\tau})$, with $\bar{\tau} \in (Const \cup Var)^k$, where k is the arity of R. *Ground literals over Const*, denoted by $\mathcal{G}(Const)$, are literals without variables, with all constants in *Const*. If $\ell = \neg R(\bar{\tau})$ then $\neg\ell \stackrel{\text{def}}{=} R(\bar{\tau})$.

Though we use the classical first-order syntax, the presented semantics substantially differs from the classical one. Namely,

- truth values $\mathfrak{t}, \mathfrak{i}, \mathfrak{u}, \mathfrak{f}$ (true, inconsistent, unknown, false) are explicitly present;[2]
- the semantics is based on sets of ground literals rather than on relational structures.

This allows one to deal with the lack of information as well as inconsistencies. Moreover, 4QL can immediately be used as the implementation tool.

The semantics of propositional connectives is summarized in Table 1. Observe that definitions of $\wedge$ and $\vee$ reflect minimum and maximum w.r.t. the ordering:

$$\mathfrak{f} < \mathfrak{u} < \mathfrak{i} < \mathfrak{t}, \tag{1}$$

as advocated, e.g., in [4,16,17,20]. Such an ordering is quite natural as truth ordering. It indicates how "true" a given proposition is. The value $\mathfrak{f}$ indicates that the proposition is definitely not true, $\mathfrak{u}$ admits a possibility that the proposition is true, $\mathfrak{i}$ shows that there is at least one witness/evidence indicating the truth of the proposition, and finally, $\mathfrak{t}$ expresses that the proposition is definitely true. Note that (1) linearizes the truth ordering of [2], where $\mathfrak{u}$ and $\mathfrak{i}$ are incomparable. We order them using knowledge ordering of [2], in which $\mathfrak{u} < \mathfrak{i}$. The implication $\rightarrow$, a four-valued extension of the classical implication, is motivated and discussed in [16,17,20].

Table 1. Truth tables for $\wedge$, $\vee$, $\rightarrow$ and $\neg$ (see [20,16,17])

$\wedge$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{t}$
$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{f}$
$\mathfrak{u}$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{u}$	$\mathfrak{u}$
$\mathfrak{i}$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{i}$
$\mathfrak{t}$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{t}$

$\vee$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{t}$
$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{t}$
$\mathfrak{u}$	$\mathfrak{u}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{t}$
$\mathfrak{i}$	$\mathfrak{i}$	$\mathfrak{i}$	$\mathfrak{i}$	$\mathfrak{t}$
$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$

$\rightarrow$	$\mathfrak{f}$	$\mathfrak{u}$	$\mathfrak{i}$	$\mathfrak{t}$
$\mathfrak{f}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$
$\mathfrak{u}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$
$\mathfrak{i}$	$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{t}$	$\mathfrak{f}$
$\mathfrak{t}$	$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{t}$	$\mathfrak{t}$

	$\neg$
$\mathfrak{f}$	$\mathfrak{t}$
$\mathfrak{u}$	$\mathfrak{u}$
$\mathfrak{i}$	$\mathfrak{i}$
$\mathfrak{t}$	$\mathfrak{f}$

Let $v : Var \longrightarrow Const$ be a *valuation of variables*. For a literal ℓ, by $\ell(v)$ we understand the ground literal obtained from ℓ by substituting each variable x occurring in ℓ by constant $v(x)$.

Definition 4.1. The *truth value* of a literal ℓ w.r.t. a set of ground literals L and valuation v, denoted by $\ell(L, v)$, is defined as follows:

$$\ell(L,v) \stackrel{\text{def}}{=} \begin{cases} \mathfrak{t} \text{ if } \ell(v) \in L \text{ and } (\neg\ell(v)) \notin L; \\ \mathfrak{i} \text{ if } \ell(v) \in L \text{ and } (\neg\ell(v)) \in L; \\ \mathfrak{u} \text{ if } \ell(v) \notin L \text{ and } (\neg\ell(v)) \notin L; \\ \mathfrak{f} \text{ if } \ell(v) \notin L \text{ and } (\neg\ell(v)) \in L. \end{cases} \qquad \lhd$$

[2] For simplicity we use the same symbols to denote truth constants and corresponding truth values.

Example 4.2. Consider the situation described in Section 3 and let $v(x) = low$. Then, for example, $P(a, x)(C_p, v) = \mathfrak{t}$ and $P(a, x)(C_e, v) = \mathfrak{u}$. $\triangleleft$

For a formula $\alpha(x)$ with a free variable x and $c \in Const$, by $\alpha(x)^x_c$ we understand the formula obtained from α by substituting all occurrences of x by c. Definition 4.1 is extended to all formulas in Table 2, where α denotes a first-order formula, v is a valuation of variables, L is a set of ground literals, and the semantics of propositional connectives appearing at righthand sides of equivalences is given in Table 1.

Table 2. Semantics of first-order formulas

- if α is a literal then $\alpha(L, v)$ is defined in Definition 4.1;
- $(\neg\alpha)(L, v) \stackrel{\text{def}}{=} \neg(\alpha(L, v))$, where $\neg$ at the righthand side of equality is defined in Table 1;
- $(\alpha \circ \beta)(L, v) \stackrel{\text{def}}{=} \alpha(L, v) \circ \beta(L, v)$, where $\circ \in \{\vee, \wedge, \rightarrow\}$;
- $(\forall x \alpha(x))(L, v) = \min_{a \in Const} (\alpha^x_a)(L, v)$, where min is the minimum w.r.t. ordering (1);
- $(\exists x \alpha(x))(L, v) = \max_{a \in Const} (\alpha^x_a)(L, v)$, where max is the maximum w.r.t. ordering (1).

Let us now define belief structures based on sets of literals. In this context a concept of an epistemic profile is the key abstraction involved in belief formation.

If S is a set then by $\text{FIN}(S)$ we understand the set of all finite subsets of S.

Definition 4.3. Let $\mathbb{C} \stackrel{\text{def}}{=} \text{FIN}(\mathcal{G}(Const))$ be the set of all finite sets of ground literals over the set of constants *Const*. Then:

- by a *constituent* we understand any set $C \in \mathbb{C}$;
- by an *epistemic profile* we understand any function $\mathcal{E} : \text{FIN}(\mathbb{C}) \longrightarrow \mathbb{C}$;
- by a *belief structure over an epistemic profile* $\mathcal{E}$ we mean $\mathcal{B}^{\mathcal{E}} = \langle \mathcal{C}, F \rangle$, where:
  - $\mathcal{C} \subseteq \mathbb{C}$ is a nonempty set of constituents;
  - $F \stackrel{\text{def}}{=} \mathcal{E}(\mathcal{C})$ is the *consequent* of $\mathcal{B}^{\mathcal{E}}$. $\triangleleft$

Example 4.4. For the example introduced in Section 3,

$\mathcal{C} = \{C_p, C_n, C_e\}$ and $F = \{\neg S(a, healthy), S(a, healthy)\}$,

so $\mathcal{E}$ is any function of the signature required in Definition 4.3 such that $\mathcal{E}(\mathcal{C}) = F$. $\triangleleft$

Definition 4.5. Let $\mathcal{E}$ be an epistemic profile. The *truth value of formula* α w.r.t. belief structure $\mathcal{B}^{\mathcal{E}} = \langle \mathcal{C}, F \rangle$ and valuation v, denoted by $\alpha(\mathcal{B}^{\mathcal{E}}, v)$, is defined by:[3]

$$\alpha(\mathcal{B}^{\mathcal{E}}, v) \stackrel{\text{def}}{=} \alpha(\bigcup_{C \in \mathcal{C}} C, v). \qquad \triangleleft$$

[3] Since $\bigcup_{C \in \mathcal{C}} C$ is a set of ground literals, $\alpha(\mathcal{S}, v)$ is well-defined by Table 2.

Example 4.6. Consider again the situation described in Section 3 and let $v(x) = low$. Let the belief structure $\mathcal{B}^{\mathcal{E}}$ be as described in Example 4.4. Then,

$$\bigcup_{C \in \mathcal{C}} = \{P(a, low), N_1(a, high), \neg Cl(a, noisy), Cl(a, neutral), Cl(a, pollutive)\}.$$

Therefore, e.g., $(P(a,x) \wedge N_1(a,x))(\mathcal{B}^{\mathcal{E}}, v) = \mathfrak{u}$ and $(P(a,x) \vee N_1(a,x))(\mathcal{B}^{\mathcal{E}}, v) = \mathfrak{t}$. ◁

To express beliefs, we extend the language with operator $\mathrm{Bel}()$ standing for beliefs. The truth table for $\mathrm{Bel}()$ is:

$$\mathrm{Bel}(\mathfrak{t}) \stackrel{\text{def}}{=} \mathfrak{t}, \quad \mathrm{Bel}(\mathfrak{i}) \stackrel{\text{def}}{=} \mathfrak{i}, \quad \mathrm{Bel}(\mathfrak{u}) \stackrel{\text{def}}{=} \mathfrak{f}, \quad \mathrm{Bel}(\mathfrak{f}) \stackrel{\text{def}}{=} \mathfrak{f}. \tag{2}$$

We say that a formula is $\mathrm{Bel}()$*-free* if it contains no occurrences of $\mathrm{Bel}()$ operator.

Definition 4.7. Let $\mathcal{E}$ be an epistemic profile. The *truth value of formula α w.r.t. belief structure $\mathcal{B}^{\mathcal{E}} = \langle \mathcal{C}, F \rangle$ and valuation v*, denoted by $\alpha(\mathcal{B}^{\mathcal{E}}, v)$, is defined as follows:

- when α is $\mathrm{Bel}()$-free then:
  - $\alpha(\mathcal{B}^{\mathcal{E}}, v)$ is defined by Definition 4.5;
  - $\mathrm{Bel}(\alpha)(\mathcal{B}^{\mathcal{E}}, v) \stackrel{\text{def}}{=} \mathrm{Bel}(\alpha(F, v))$, where the truth value $\alpha(F, v)$ is defined in Table 2 and $\mathrm{Bel}()$ applied to a truth value is defined by (2);
- when $\mathrm{Bel}()$ operators are nested in α then $\alpha(\mathcal{B}^{\mathcal{E}}, v)$ is evaluated starting from the innermost occurrence of $\mathrm{Bel}()$, which is then replaced by the obtained truth value, etc. ◁

Example 4.8. For the belief structure $\mathcal{B}^{\mathcal{E}}$ introduced in Section 3 (see also Example 4.4) and $v(x) = low$, we have:

$$\begin{aligned}&\big(P(a,x) \wedge \mathrm{Bel}(P(a,x) \vee \mathrm{Bel}(S(a, healthy)))\big)(\mathcal{B}^{\mathcal{E}}, v) = \\ &\quad \mathfrak{t} \wedge \mathrm{Bel}(P(a,x) \vee \mathrm{Bel}(\mathfrak{i})) = \mathfrak{t} \wedge \mathrm{Bel}(P(a,x) \vee \mathfrak{i}) = \mathfrak{t} \wedge \mathrm{Bel}(\mathfrak{u} \vee \mathfrak{i}) = \\ &\quad \mathfrak{t} \wedge \mathrm{Bel}(\mathfrak{i}) = \mathfrak{t} \wedge \mathfrak{i} = \mathfrak{i}.\end{aligned}$$
◁

One can easily verify the following proposition.

Proposition 4.9. *For any formula α, belief structure $\mathcal{B}^{\mathcal{E}}$ and valuation of variables v:*

$$\big(\neg \mathrm{Bel}(\mathfrak{f})\big)\big(\mathcal{B}^{\mathcal{E}}, v\big) = \mathfrak{t}; \tag{3}$$

$$\big(\mathrm{Bel}(\alpha) \to \mathrm{Bel}(\mathrm{Bel}(\alpha))\big)\big(\mathcal{B}^{\mathcal{E}}, v\big) = \mathfrak{t}; \tag{4}$$

$$\big(\neg \mathrm{Bel}(\alpha) \to \mathrm{Bel}(\neg \mathrm{Bel}(\alpha))\big)\big(\mathcal{B}^{\mathcal{E}}, v\big) = \mathfrak{t}. \tag{5}$$

Observe that above formulas express the classical properties of beliefs: (3) is the axiom **D**, (4) and (5) are axioms **4** and **5**, expressing positive and negative introspection. Note that modal logic **KD45**, based on these axioms, is typically used to model beliefs in multiagent systems. Furthermore, there are belief structures, where the following axiom **T**, distinguishing knowledge and beliefs, does not have to be true:

$$\mathrm{Bel}(\alpha) \to \alpha. \tag{6}$$

This follows from the fact that given a belief structure $\langle \mathcal{C}, F\rangle$, $\mathrm{Bel}(\alpha)$ evaluates α in F while α itself is evaluated in $\bigcup \mathcal{C}$.

Let us also note that the following axiom:

$$\neg\big(\mathrm{Bel}(\alpha) \wedge \mathrm{Bel}(\neg\alpha)\big), \tag{7}$$

sometimes replacing axiom (3) is not always true under our semantics. For example, when α is $\mathfrak{i}$ then formula (7) is $\mathfrak{i}$. Axioms (3) and (7) are equivalent in the context of **KD45**. However, this is no longer the case in our semantics as we allow agents to have inconsistent beliefs.

5 Pragmatics

5.1 Individual Beliefs

Agents can acquire knowledge about other agents' beliefs via communication and observation. In contrast to many existing approaches, we do not assume that an agent entering a group changes its beliefs accordingly. For example, when two agents cooperate, they may have certain beliefs as a group, but do not have to revise their individual beliefs. This usually results in a substantial complexity improvement. When the group is dismissed, agents continue to act according to their individual beliefs. When agents cooperate, a specific group is, possibly implicitly, created, including an epistemic profile fitting the entire situation. This is where belief fusion methods adequate for the group in question occur. In general, any interaction between agents leads to the creation of a (possibly virtual) group with a specific epistemic profile. We can naturally model this process in the proposed framework.

Let $\{Ag_i \mid i = 1, \ldots, n\}$ be a *set of agents*. To model individual beliefs we introduce belief operators $\mathrm{Bel}_i(\alpha)$, for $i = 1, \ldots, n$. As usually, the formula $\mathrm{Bel}_i(\alpha)$ expresses that agent Ag_i believes in α. To define the semantics of $\mathrm{Bel}_i(\alpha)$ operators we assume that for $i = 1, \ldots, n$, $\mathcal{E}_i$ is an epistemic profile of agent Ag_i and $\mathcal{B}^{\mathcal{E}_i} = \langle \mathcal{C}_i, F_i\rangle$ is a belief structure of agent Ag_i.

Definition 5.1. Let $\bar{\mathcal{B}} = \{\mathcal{B}^{\mathcal{E}_i} \mid i = 1, \ldots, n\}$ be a tuple of belief structures. The *truth value of formula α w.r.t. $\bar{\mathcal{B}}$ and valuation v w.r.t. agent Ag_i*, denoted by $\alpha(i, \bar{\mathcal{B}}, v)$, is defined as follows:

– when α is $\mathrm{Bel}()$-free then:
  - $\alpha(i, \bar{\mathcal{B}}, v)$ is defined as $\alpha(\mathcal{B}^{\mathcal{E}_i}, v)$ in the sense of Definition 4.5;
  - $\mathrm{Bel}_j(\alpha)(i, \mathcal{B}^{\mathcal{E}}, v) \stackrel{\mathrm{def}}{=} \mathrm{Bel}(\alpha(F_j, v))$, where the truth value $\alpha(F_i, v)$ is defined in Table 2 and $\mathrm{Bel}()$ applied to a truth value is defined by (2);

– when $\mathrm{Bel}_j()$ operators are nested in α then $\alpha(i, \bar{\mathcal{B}}, v)$ is evaluated starting from the innermost occurrence of $\mathrm{Bel}()$, which is then replaced by the obtained truth value, etc. ◁

Example 5.2. When agent A_k evaluates formula $\big(r \vee \mathrm{Bel}_i(\mathrm{Bel}_j(p) \wedge q)\big)$ w.r.t. v then:

– r is evaluated w.r.t. F_k and v;
– p in $\mathrm{Bel}_j(p)$ is evaluated w.r.t. F_j and v;
– q in $\mathrm{Bel}_i(\mathrm{Bel}_j(p) \wedge q)$ is evaluated w.r.t. F_i and v. ◁

5.2 Group Beliefs

A group of agents, say $G = \{Ag_{i_1}, \ldots, Ag_{i_j}\}$, has its *group belief structure* $\mathcal{B}^{\mathcal{E}_G} = \langle \mathcal{C}_G, F_G \rangle$, where $\mathcal{C}_G = \{F_{i_1}, \ldots, F_{i_j}\}$. Thus, consequents of group members become constituents of a group. The group then builds group beliefs via its epistemic profile $\mathcal{E}_G$, e.g. by adjudicating beliefs of group members, and reaches its consequent F_G (see Figure 1).

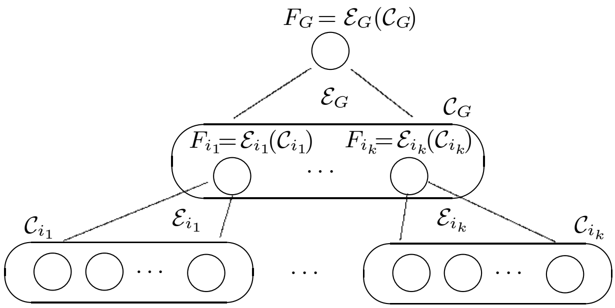


Fig. 1. The architecture of individual and group beliefs

To express properties of group beliefs we extend the language with operators $\text{Bel}_G(\alpha)$, where G is a group of agents.

Let $\{Ag_i \mid i = 1, \ldots, n\}$ be a *set of agents* and $\{G_j \mid j = n+1, \ldots m\}$ be a set of groups of agents. To define the semantics of $\text{Bel}_G(\alpha)$ operators we extend Definition 5.1 by assuming that for $l = n+1, \ldots, m$, $\mathcal{E}_l$ is an epistemic profile of group G_l and $\mathcal{B}^{\mathcal{E}_l} = \langle \mathcal{C}_l, F_l \rangle$ is a belief structure of group G_l. We therefore have a tuple of belief structures $\bar{\mathcal{B}} = \{\mathcal{B}^{\mathcal{E}_l} \mid l = 1, \ldots, n\}$, where for $i = 1, \ldots, n$, $\mathcal{B}^{\mathcal{E}_i}$ is a belief structure of agent Ag_i and for $j = n+1, \ldots, m$, $\mathcal{B}^{\mathcal{E}_j}$ is a belief structure of group G_j.

Since groups are dealt with exactly as agents, given a tuple of belief structures $\bar{\mathcal{B}} = \{\mathcal{B}^{\mathcal{E}_l} \mid l = 1, \ldots, n\}$, the truth value of formula α w.r.t. $\bar{\mathcal{B}}$ and valuation v w.r.t. agent Ag_i (respectively, group G_i), is defined exactly as in Definition 5.1, assuming that indices $1, \ldots, n$ refer to agents and indices $n+1, \ldots, m$ refer to groups.

The same way beliefs of groups involving other groups may be formed. For example, a surveillance group of robots G_s may join a rescue team of robots G_r making a larger group $G_{s,r}$. Then the consequents of G_s and G_r become constituents of $G_{s,r}$ and the first step is completed. Furthermore, such groups can become parts of other, more complex groups, and so on. The underlying methods for forming group beliefs on the top of group members' beliefs are typically highly application- and context-dependent.

To express beliefs of such groups we extend the language with $\text{Bel}_{\mathbb{G}}()$ operators, where $\mathbb{G}$ may contain individual agents, groups of agents, groups of groups of agents, etc. Since such complex groups, when formed, are equipped with belief structures like in the case of groups consisting of agents only, the semantics of $\text{Bel}_{\mathbb{G}}()$ operators is given by an immediate adaptation of the semantics of groups.

Note that, due to complexity reasons, it is reasonable to assume that only formed groups are equipped with belief structures and epistemic profiles. When a group is not formed, we assume that its belief structure $\mathcal{B}$ is "empty", i.e., $\mathcal{B} = \langle \mathcal{C}, F \rangle$ with $\mathcal{C} = \{\emptyset\}$ (that is, $\mathcal{C}$ consists of a single set being the empty set) and $F = \emptyset$. Note that all queries supplied to this structure return the value $\mathfrak{u}$.

5.3 Feasibility of the Approach

As already indicated, all epistemic profiles and belief structures implemented in 4QL are tractable. Moreover, all epistemic profiles and belief structures constructible in polynomial time can be implemented in 4QL. On the other hand, given n agents, one can have $(2^n - 1)$ nonempty groups of agents, so there is a natural question about the feasibility of the proposed framework in real-world agent systems. It should be noted, however, that in practice such groups are dynamically generated at runtime, so whenever in a particular application there is a large number of groups, it means that the computational power used has been strong enough to generate an maintain them.

Agent's and groups' belief structures and epistemic profiles typically match some patterns reflecting agents' types or groups' organizational structures and cooperation procedures. Therefore, in practice, one can expect belief structures to be generated on the basis of a library of patterns, much like in object-oriented programming dynamic objects are generated on the basis of static classes, developed during the system's design phase. Of course, the number of such patterns does not change during system execution, so can be bound by a constant.

6 Conclusions

In this paper we succeeded to differentiate agent's characteristics via individual and group epistemic profiles, reflecting agent's reasoning capabilities. This abstraction tool permits both to flexibly define the way an agent (a group) reasons and to reflect the granularity of reasoning.

The presented novel and pragmatic framework to beliefs suits real-world applications that often are not easy to formalize. In particular, it allows for natural handling of inconsistencies and gaps in beliefs by using paraconsistent and nonmonotonic reasoning.

Moreover, our approach permits a uniform modeling of individual and group beliefs, where group is a generic concept consisting of individual agents, groups of agents, groups of groups of agents, etc. Importantly, the assumed layered architecture underlying the framework allows one to avoid costly revisions of agents' beliefs when they join a group. This is especially important when paradigmatic agent interaction is considered. Cooperation, coordination and communication is naturally modeled by creating a group and forming group beliefs to achieve a common informational stance. What sort of structure it is and how this influences agents' individual beliefs is a matter of design decisions. Our approach, ensures both the heterogeneity of agents involved and a flexibility of group level reasoning patterns.

Most significantly, we have indicated 4QL as a tool to implement all epistemic profiles and belief structures constructible in deterministic polynomial time. One can then query them in a tractable manner, which provides a rich but still pragmatic reasoning machinery. To the best of our knowledge, such tractability of reasoning about beliefs has not been achieved yet. Also, nonmonotonic/defeasible reasoning techniques are easily expressible in 4QL ensuring both richness and flexibility of implemented epistemic profiles.

All these aspects call for extensive case studies as the one shown in [6]. Another interesting problem for future work is to provide a sound and complete proof system for verifying properties of beliefs.

References

1. Ågotnes, T., Alechina, N.: Full and relative awareness: A decidable logic for reasoning about knowledge of unawareness. In: Proc. of TARK, pp. 6–14. ACM Press (2007)
2. Belnap, N.: A useful four-valued logic. In: Eptein, G., Dunn, J. (eds.) Modern Uses of Many Valued Logic, pp. 8–37. Reidel (1977)
3. Béziau, J.-Y., Carnielli, W., Gabbay, D. (eds.): Handbook of Paraconsistency. College Publications (2007)
4. de Amo, S., Pais, M.: A paraconsistent logic approach for querying inconsistent databases. International Journal of Approximate Reasoning 46, 366–386 (2007)
5. Dubois, D.: On ignorance and contradiction considered as truth-values. Logic Journal of the IGPL 16(2), 195–216 (2008)
6. Dunin-Kęplicz, B., Szałas, A.: Implementing belief structures and epistemic profiles: A case study (submitted 2012)
7. Dunin-Kęplicz, B., Verbrugge, R.: Teamwork in Multi-Agent Systems. A Formal Approach. John Wiley & Sons, Ltd. (2010)
8. Dunin-Kęplicz, B., Verbrugge, R.: Awareness as a vital ingredient of teamwork. In: Stone, P., Weiss, G. (eds.) Proc. of AAMAS 2006, pp. 1017–1024 (2006)
9. Fagin, R., Halpern, J.: Belief, awareness, and limited reasoning. Artificial Intelligence 34(1), 39–76 (1988)
10. Fagin, R., Halpern, J., Moses, Y., Vardi, M.: Reasoning About Knowledge. The MIT Press (2003)
11. Gottlob, G.: Complexity results for nonmonotonic logics. Journal of Logic and Computation 2(3), 397–425 (1992)
12. Hintikka, J.: Knowledge and Belief. Cornell University Press (1962)
13. Kraus, S., Lehmann, D.: Knowledge, belief, and time. Theoretical Computer Science 58, 155–174 (1988)
14. Laux, A., Wansing, H. (eds.): Knowledge and Belief in Philosophy and Artificial Intelligence. Akademie Verlag, Berlin (1995)
15. Łukaszewicz, W.: Non-Monotonic Reasoning - Formalization of Commonsense Reasoning. Ellis Horwood (1990)
16. Małuszyński, J., Szałas, A.: Living with Inconsistency and Taming Nonmonotonicity. In: de Moor, O., Gottlob, G., Furche, T., Sellers, A. (eds.) Datalog 2010. LNCS, vol. 6702, pp. 384–398. Springer, Heidelberg (2011)
17. Małuszyński, J., Szałas, A.: Logical foundations and complexity of 4QL, a query language with unrestricted negation. Journal of Applied Non-Classical Logics 21(2), 211–232 (2011)
18. Marek, V., Truszczyński, M.: Nonmonotonic Logic. Springer (1993)
19. van Ditmarsch, H., van der Hoek, W., Kooi, B.: Dynamic Epistemic Logic. Springer (2007)
20. Vitória, A., Małuszyński, J., Szałas, A.: Modeling and reasoning with paraconsistent rough sets. Fundamenta Informaticae 97(4), 405–438 (2009)

To a Formal Modeling Approach of Error Recovery in Production Systems Based on Holonic Multi-agent Systems Specification

Belhassen Mazigh[1,*], Ahmed Hammed[1], and Vincent Hilaire[2]

[1] Research Unit MARS, Faculty of Sciences, 5000, Monastir, Tunisia
[2] Laboratoire Systèmes Et Transport, Université de Technologie de Belfort-Montbéliard, Rue Thierry Mieg, 90010 Belfort cedex, France
belhassen.mazigh@gmail.com, ahmed.hammed@yahoo.fr, vincent.hilaire@utbm.fr

Abstract. Production systems must be reconfigurable and endowed with methods and tools allowing an automatic recovery of expected and unexpected errors. In this domain, error recovery in production systems is always an open problem. Multi-agent systems seem to be adapted for representing this kind of problem in an attempt to arrive at a reconfigurable, adaptive, and "intelligent" manufacturing systems. The objective of this work consists in proposing a specification, modeling and structural analysis approach of error recovery production systems using holonic multi-agent systems and Object Petri Nets (OPN).

Keywords: Error Recovery, Holonic Multi-Agent Systems, Object Petri Nets, Production Systems, Structural Analysis.

1 Introduction

Researches on intelligent agents in the context of manufacturing have been mostly concentrated on the "production activities" e.g. scheduling, planning, processing and material handling. However the activities related to exception handling such as diagnostics and error recovery have received little attention. Our research aims to propose a holonic multi-agent based approach from the specification using ASPECS [1] until the formal modeling and structural check using object Petri nets of error recovery system.

After a brief presentation of the error recovery problem, an error recovery system context is presented. In section 3, the statement of the problem is presented. A quick overview of the ASPECS process and modeling approach will be presented in section 4. The analysis and conception phase of the ASPECS process and their associated activities are then described in Section 5. This section also presents the application of ASPECS activities to design error recovery system. Section 6 presents the formal specification of the error recovery system with object Petri nets using the renew editor. Section 7 consists in the application of error recovery model on the robotic cell for piston assembly example. Section 8 summarizes the results of the paper and describes some future work directions.

* Corresponding author.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 370–378, 2012.

2 Error Recovery System Context

One of the major problems in manufacturing systems is how to effectively recover from both anticipated and unanticipated faults. Traditional techniques have addressed the error recovery problem from the point of view of defining a set of actions for a known set of errors. In this context, the authors of [2] proposed an error recovery approach based on the integration of explanation-based and analogy-based learning. The knowledge representation for the proposed learning technique is based on the object-oriented modeling concept. Zhou and DiCesare [3] developed a formal description of four possible error recovery strategies in terms of Petri net constructs, namely input conditioning, backward error recovery, forward error recovery and alternate path recovery. The authors of [4] proposed a strategy for online recovering of the manufacturing operationality by using a multi-agent based decision support system with fuzzy reasoning.

In an attempt to investigate an “intelligent” manufacturing workstation controller, an approach integrating Petri net models and neural network techniques for preliminary diagnosis was undertaken [5]. More recent works pertain to addressing the issue of monitoring, diagnostics, and error recovery within the context of hierarchical multi-agent systems [6].

3 Statement of the Problem

The focus in this research is on recovery of physical error. The chosen approach is to design a reconfigurable, adaptive and intelligent manufacturing system. As such, a specification approach based on holonic multi-agent systems seems to be a promising approach to deal with the unpredictable nature of error due to their decentralization, autonomy, cooperation features and their hierarchical ability to react to unexpected situation. For this specification, we use the agent-oriented software process called ASPECS which is based on a holonic organizational metamodel and provides a step-by-step guide from requirements to code allowing the modeling of a system at different levels of details using a set of refinement methods. We use this methodology to identify holonic organization of a steady system and we propose a Petri Net approach for modeling and analysis specification associated to the case steady.

4 A Quick Overview of the Used Aspecs Process

As mentioned in [7], the ASPECS Process is a step-by-step requirement to code software engineering process based on a metamodel, which defines the main concepts for the proposed HMAS analysis, design and development [8][9].

The target scope for the proposed approach can be found in complex systems and especially hierarchical complex systems. The main vocation of ASPECS is towards the development of societies of holonic (as well as not holonic) multi-agent systems. ASPECS has been built by adopting the Model Driven Architecture (MDA) [10]. In Cossentino and al. [11] they label the three metamodels “domains” thus maintaining the link with the PASSI metamodel. The three definite fields are:

- The Problem Domain. It provides the organizational description of the problem independently of a specific solution. The concepts introduced in this domain are mainly used during the analysis phase and at the beginning of the design phase.
- The Agency Domain. It introduces agent-related concepts and provides a description of the holonic, multi-agent solution resulting from a refinement of the Problem Domain elements.
- The Solution Domain is related to the implementation of the solution on a specific platform. This domain is thus dependent on a particular implementation and deployment platform.

Our contribution will relate to the consolidation of the Problem Domain and the Agency Domain. We propose a formal specification approach for the analysis of the various organizations and the interactions between them facilitating therefore the Solution Domain.

5 Holarchy Design of Error Recovery System (ERS)

In this section, we use the ASPECS methodology to describe partially the analysis phase, the design of the agent society and to propose a holonic structure of Error Recovery System (Fig.1).

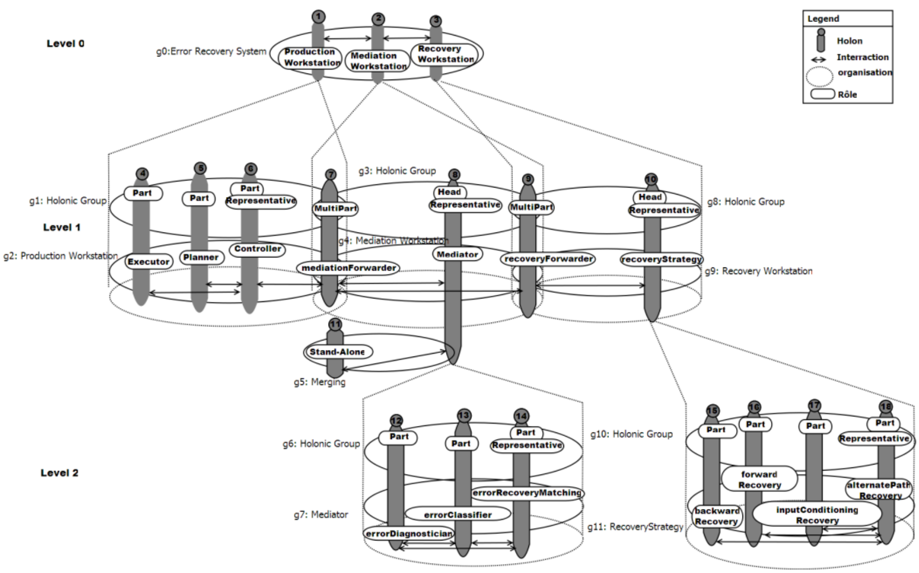


Fig. 1. Holarchy of Error Recovery System

At the level 0 of the Holarchy, three super-holons H1, H2 and H3 play respectively the role of the “Production Workstation”, ”Mediation Workstation” and “Recovery Workstation” in g0 group. H1 contains an instance of “Production Workstation” organization (g2), H2 contains an instance of “Mediation Workstation” (g4) and H3 contains an instance of “Recovery Workstation” (g9). The super-holon H8 of the “Mediation Workstation” Organization plays the role of the “mediator” in g4 group (Mediation Workstation). It contains an instance of “mediator” organization.

The super-holon H10 of the “Recovery Workstation” Organization plays the role of the “RecoveryStrategy” in g9 group (”Recovery Workstation”). It contains an instance of “RecoveryStrategy” organization. The holon H6 playing the role of the “Controller” in the “Production Workstation” organization, the two super-holons H8and H10, the holon H14 playing the role of the “errorRecoveryMatching” in the “mediation” organization and the holon H18 playing the role of the “alternatePathRecovery” in the “Recovery strategy” organization named Head and Representative of the other members. The holon H7 playing the role of the “mediationForwarder” and the holon H9 laying the role “recoveryForwarder” are named Multi-part as its shared their roles respectively between “Production Workstation” and “Mediation Workstation” organizations and ”Mediation Workstation” and “Recovery Worksation” organizations.

Concerning H11, it’s a Stand-Alone holon. In fact, when the holon H8 will diagnostic the error at the aim to classify it and to choose the adequate recovery policy, this error may be unanticipated and by consequent, the recovery method is unknown.

In this case, H11 is called to find a solution to this error, it can for example try all the recovery policy presented or request a sub-contracted service.

6 Formal Modeling

The focus in this section is on the formal modeling of the error recovery system. Petri nets are well suited for use in formal modeling of multi-agent systems. In this context, Mulan [12] (Multi-Agent Nets) is an architecture used for the multi-agent systems. It‘s built on Java and reference nets and can be executed in Renew [13]. However, there exists no released Version of Mulan that allows us to model multi-agent systems in this architecture.

On the other hand, OREDI [14] (ORganization EDItor), a Petri net based tool enables editing organization models as well as deploying such models as multi-agent systems. OREDI is built on top of Renew and relies on SONAR [15] (Self Organizing Net architecture reference), a formal organizational specification for electronic institutions based on Petri nets. However, OREDI has not been released yet.

For the formal modeling of our error recovery system, we use object Petri net with Renew editor. Renew is a high-level Petri net editor and simulator that achieves a seamless integration of Petri nets and the Java programming language. Petri nets can implement methods and can be treated as first-class Java objects. On the other hand, Java code can be accessed from nets easily.

Renew implements bidirectional synchronous channels and multiple instances of a single net, there, by providing powerful abstraction concepts. The simulation engine is capable of executing multiple transitions in separate threads. Renew is well suited for rapid prototyping and creating executable workflow models.

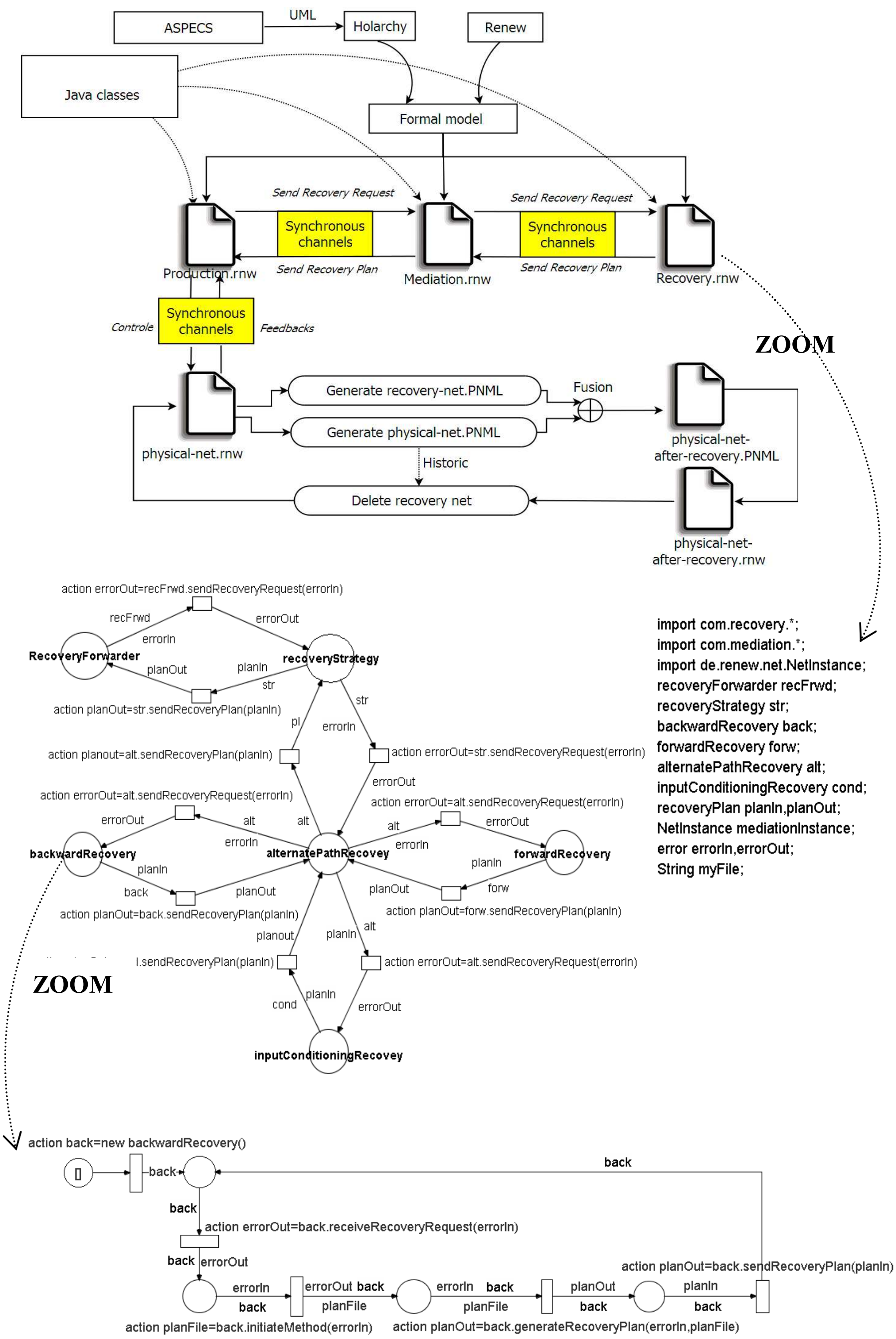


Fig. 2. Formal model of Error Recovery System

The formal model will be deduced from the holarchy presented in the previous section. In fact, the three base organizations: Production Workstation, Mediation Workstation and Recovery Workstation will be represented by three rnw files (rnw: extension of Renew files) and they will be described respectively by the Java packages Production, Mediation and Recovery. Agents are represented by Java classes of which the attributes are the characteristics of these agents and methods are actions which can be executed by these agents.

Classes belonging to the same package represent agents of the same organization. Interactions between organizations are modeled by synchronous channels implemented in Renew. The Fig. 2 describes our formal model for the error recovery system.

7 Verification and Validation of the Model

In this section, we try to verify and validate our formal model on a robotic cell for piston assembly workstation example taken from Zhou and DiCesare‘s book [3]. The workstation uses two robots to place and pull the pistons into the cylinders of an engine block and attaches the piston rods to the crank shaft.

Fig. 3 demonstrates necessary steps to apply our error recovery model on this example (robotic cell for piston assembly) from error detection to the recovery of this error and the return back of the system to his normal state.

In this example, we treat three types of error with three different recovery policies as mentioned by Zhou and DiCesare: piston-puller-tool-down error with input conditioning error recovery, out-of-tolerance error with alternate path error recovery and incorrect part orientation error using the forward error recovery method.

When an error is detected, only its specification and its error occurrence place are sent by the « Executor » agent to the « Controller » which assigns to this error an identifier “errorId” and forward it to the “Mediation Workstation”.

The error is diagnosed by the “errorDiagnostician” agent, classified by the “errorClassified” agent and an error recovery policy is chosen by the “errorRecoveryMatching” agent. After that, the error is forwarded to the “Recovery Workstation” when a recovery plan is generated by the corresponding error recovery agent and sent to the “Production Workstation” via the “Mediation Workstation”.

The “Planner” agent transforms the recovery plan to a PNML file and adds it to the PNML file of the physical system. Finally, the “Executor” agent generates a Petri net corresponding to the physical system and the recovery module which will be deleted when the recovery procedure is achieved.

Until achieving this work, Renew does not allow to make structures check, but it can generate the format PNML [16] (Petri Net Markup Language) which is a proposal of an XML-based interchange format for Petri nets.

For this, and since validation of object Petri nets and colored Petri nets are the same, the idea is to transform our Petri nets modules to ordinary Petri nets and to use Tina as Petri net editor that can make structural analysis (liveness, safeness, reversibility and boundedness) and support PNML format.

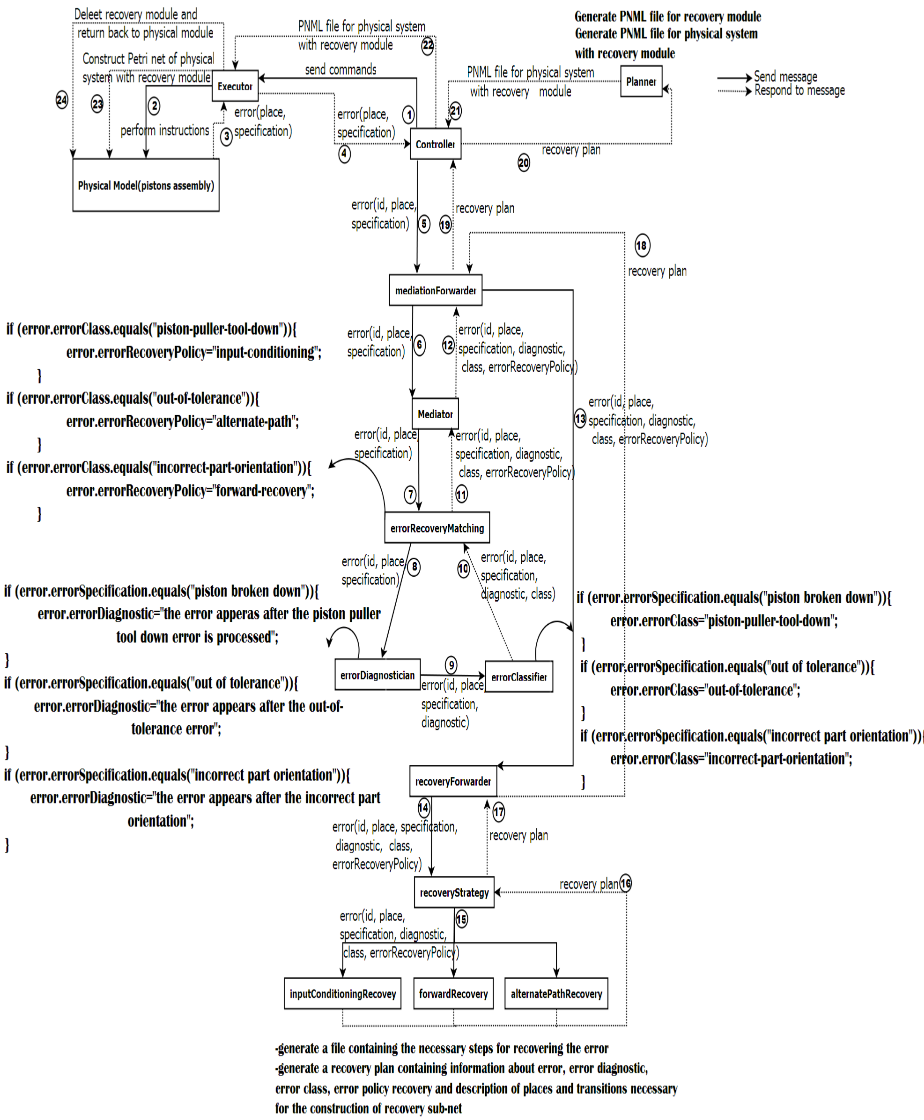


Fig. 3. The error recovery model for piston assembly workstation

The Fig. 4 represent a part of our formal model, it describes the Petri net model of the example physical system with the sub Petri net of error recovery module. The sub Petri net module describes an alternate path error recovery policy. It assumes that an out-of-tolerance condition for a nut is built up and detected in the place P16. The error recovery procedure consists of three transitions t13, t14 and t15 and three places: P21: M-1 discards the nuts, P22: Two nuts available and P23: M-1 picks up the nuts. This Petri net is live, bounded, reversible and safe. In fact, the analyze check of corresponding PNML file with Tina editor gives results described by Fig. 5:

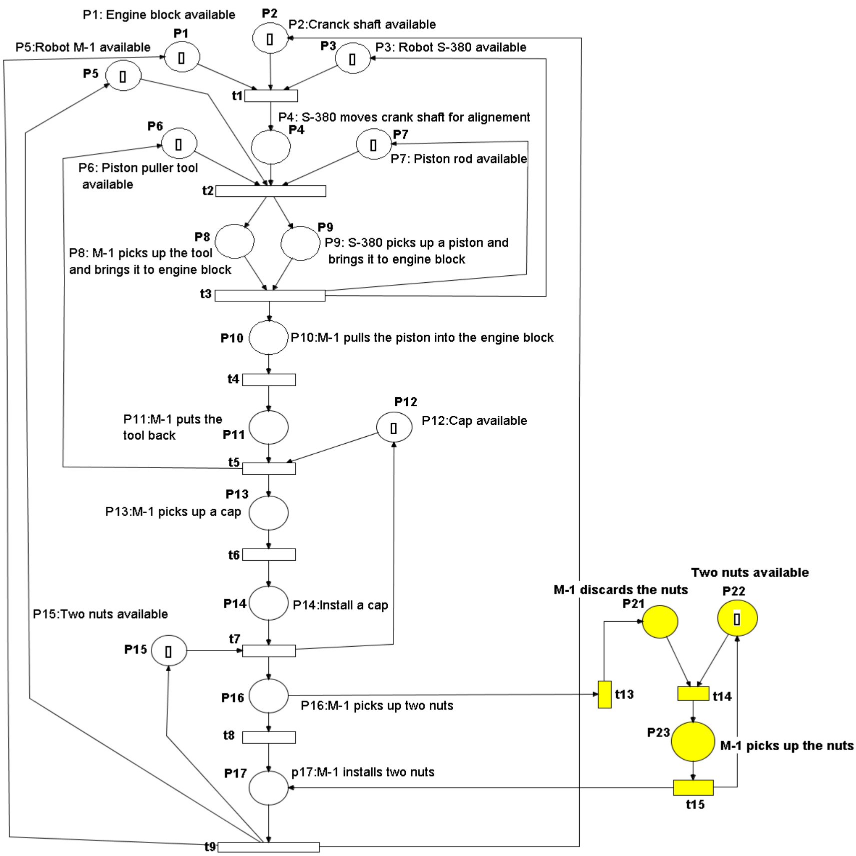


Fig. 4. Physical system model with error recovery module

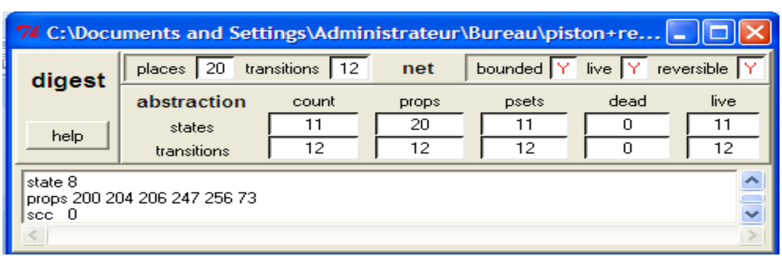


Fig. 5. Analyze check results of Physical system model with error recovery module

8 Conclusion

In this article, we presented an approach of a specification, modeling and structure check of error recovery system. This approach is based on ASPECS to design the holarchy of our system and Renew editor to construct a formal model corresponding to the holarchy using object Petri nets. Recently, a new plug in called Lola was added to Renew tool and it integrates verification capabilities of Petri nets. Although, it's an experimental plug in and ignores all java inscriptions, guards and synchronous channels. So, it's like the work that we have done with Tina. Our future works will focus on the implementation of a tool which could generate holonic multi-agent systems with objects Petri nets.

References

1. http://www.aspecs.org
2. Chang, S.J., Goldbogen, G., Dicesare, F.: Automated error recovery in manufacturing systems through learning and reasoning. In: Proceedings of the 5th IEEE International Symposium on Digital Object Identifier (1990), doi:10.1109/ISIC.1990.128578
3. Zhou, M., DiCesar, F.: Petri Net Synthesis for Discrete Event Control of Manufacturing Systems. Kluwer Academic Publishers, USA (1993)
4. Ulieru, M., Norrie, D.: A multi-agent system for on-line fault recovery of intelligent manufacturing systems. In: 18th International Conference of the North American on Digital Object Identifier Fuzzy Information Processing Society, NAFIPS (1999), doi:10.1109/NAFIPS.1999.781814
5. Ma, Y., Odrey, N.: On the application of Neural Networks to a Petri net –based intelligent workstation controller for manufacturing. In: Proceedings of the Artificial Neural Networks in Engineering (ANNIE 1996) Conference, St. Louis, MO, vol. 6, pp. 829–836. ASME Press, NY (1996)
6. Odrey, N.: Error Recovery in Production Systems: A Petri Net Based Intelligent System Approach, in. In: Kordic, V. (ed.) Petri Net Theory and Applications, vol. 7, p. 534. I-Tech Education and Publishing, Vienna (2008) ISBN 978-3-902613-12-7
7. Mazigh, B., Hilaire, V., Koukam, A.: Formal Specification of Holonic Multi-agent Systems: Application to Distributed Maintenance Company. In: Proceeding of 9th International Conference on Practical Applications of Agents and Multiagent Systems (2011)
8. Gaud, N.: Systèmes Multi-Agent Holoniques: de l'analyse à l'implantation. PhD thesis, Université de Technologie de Belfort-Montbéliard (2007)
9. Cossentino, M., Gaud, N., Galland, S., Hilaire, V., Koukam, A.: A Holonic Metamodel for Agent-Oriented Analysis and Design. In: Mařík, V., Vyatkin, V., Colombo, A.W. (eds.) HoloMAS 2007. LNCS (LNAI), vol. 4659, pp. 237–246. Springer, Heidelberg (2007)
10. Object Management Group. MDA guide, v1.0.1, OMG/2003-06-01 (2003)
11. Cossentino, M., Gaud, N., Hilaire, V., Galland, S., Koukam, A.: ASPECS: an agent-oriented software process for engineering complex systems How to design agent societies under a holonic perspective. Auton Agent Multi-Agent Syst. 20, 260–304 (2010)
12. Köhler, M., Moldt, D., Rölke, H.: Modelling the Structure and Behaviour of Petri Net Agents. In: Colom, J.-M., Koutny, M. (eds.) ICATPN 2001. LNCS, vol. 2075, pp. 224–241. Springer, Heidelberg (2001)
13. Kummer, O., Wienberg, F., Duvigneau, F.: Renew – User Guide, University of Hamburg, Computer Science Department, Vogt-Kölln Str. 30, 22527 Hamburg, Deutschland, 1.6 edn. (2002)
14. Delui, E., Kohler, M.: SONAR/OREDI: A tool for creation and deployment of organization models, Germany (2010)
15. Kohler, M.: A formal model of multi-agent organizations. Fundamenta Informaticae 79, 415–430 (2007)
16. http://www.pnml.org

Holonification of a Network of Agents Based on Graph Theory

Monireh Abdoos, Ahmad Esmaeili, and Nasser Mozayani

Computer Engineering Department, Iran University of Science and Technology, Tehran, Iran

Abstract. A multi-agent system consists of a group of interacting autonomous agents. The key problem in such a system is coordination and cooperation, i.e. how to ensure that individual decisions of the agents result in jointly optimal decisions for the overall system. This problem becomes even more serious when the number of the agents is large. Holonic model is an effective method to manage large scale problems. In holonic approaches, the formation of the initial holons is very critical and has a great influence on their performance and effectiveness. In this paper, we use a graph based modelling approach to group a population of agents with a greedy method, driven by a very simple and effective quality measure. The proposed method is evaluated by applying it to an urban traffic problem as a case study and it is shown the proposed method produces better results.

1 Introduction

Multi-Agent Systems (MASs) have often been effective tools for simulation and modelling of distributed applications [1]. Most of real world applications are complex and large scale in which we usually find a great number of interacting entities. The problems modelled by multi-agent systems can be formulated as decision making problems where an agent needs to decide which action to execute in order to maximize its objective function. Optimization of the decision making process in multi-agent systems is very challenging due to the fact that each agent needs to take into account other agents in the system. An approach to cope with large scale systems is to organize agents towards a common goal where each agent interacts with the other agents according to a network topology [2]. In the organization paradigm interactions among the agents are optimized and reduced effectively which makes it a very popular technique to manage the complexity of large systems. In multi-agent systems, an organization is defined as the collection of roles, relationships and authority structures which govern the behaviour of the system. There are many models proposed for the concept of organizations in multi-agent system. The most commonly ones used are: hierarchies, holarchies, coalitions, teams, congregation, societies, federations, markets, matrix organization and compound organizations [3]. In this paper, we will focus on holarchies and holonic systems.

Every holonic system needs a holonification process for formation of structure of the holons [4]. In other words, the aim in holonification is to specify the way that agents organize inside of a holon and make different levels in the holarchy. Holonification

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 379–388, 2012.

is considered very critical in design of holonic multi-agent systems and an improper method may increase the complexity of the system and decrease the efficiency.

This paper presents a new holonification method that enables the agents to find and form proper holons. In the proposed method, we have made use of graphs to model the population of agents in the holonification process; in addition, a quality measure is introduced for estimating the quality of the created holons. The proposed method is applied to urban traffic problem where an attempt is made to cooperatively control traffic signals.

This paper is organized as follows: in section two we present a brief description of holonic multi-agent system. Section three explains our graph-based holonification algorithm in detail. A case study of traffic signal control using holonic approach is presented in section four. Finally, section five gives a conclusion and future works.

2 Holonic Multi-Agent System

The term ”holonic” introduced by Koestler (1967) which is a combination from the Greek holos = whole, with the suffix -on which means a particle or part [5]. A holonic organization is a hybrid, recursive and hierarchical structure which is able to generate dynamic linkages to control structure. A holon is characterized by some features: being stable, having the capability of autonomy, being capable of cooperation [6]. In a holonic system, the concept of a holarchy refers to architecture of self-organizing open hierarchical systems [5]. A holarchy can be modelled using whole-part relationships and it is managed in a distributed manner by system elements or holons. Fig. 1 shows a three-level holarchy. In Holonic Multi-Agent System (HMAS) the structure can be

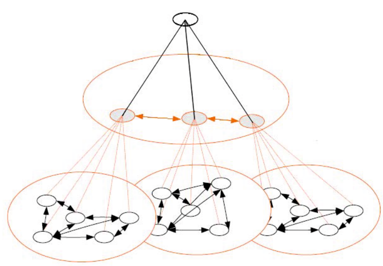


Fig. 1. Holarchy

seen as a set of hierarchical levels, where the agents can interact only with other agents at the same level or at the level immediately below or above. In these systems, individual holons define their activities based on their local knowledge and decide about their behaviour by means of the permitted interactions. Holonic organizations have proven to be effective solutions to several problems associated with hierarchical and self organizing structures [7] and have been successfully applied in a wide range of complex systems. For instance, we can mention the works done in manufacturing systems [8], transportation [9], adaptive mesh problem [10], health organizations [11].

3 Holonification

In this section, we present our graph-based approach for constructing the holarchies. The first step in the holonification process involves modelling different entities of the system and the relations among them. Then, an algorithm is needed to properly extract the structure of the holons based on the model. Furthermore, a measure should be used to evaluate the quality of the extracted holons.

3.1 Graph-Based Modelling

Graph concepts, as one of very old and effective concepts, have always been found useful in the modelling of the relationships and interactions in large and complex systems. Depending on the applications and relationships among the agents, a directed or undirected graph can be utilized. In this paper, an undirected and weighted graph is used in which, the agents are represented by the vertices and the relationships between any two agents is represented by an edge in the graph.

Holonification in the agent's network is very similar to the concept of partitioning in graph theory. In a graph partitioning problem, the aim is to divide the vertices of a graph into a set of partitions in such a way that some criteria are satisfied. Holonification is a process of partitioning the network into communities or holons in such a way that the most related group of the agents belong to the same holon. This process as like the graph partitioning process is NP-hard [12] and an algorithm to approximate a near optimal solution is desirable [13].

Formally, let's denote a graph by the pair $G =< V,E >$ in which V and E are the sets of vertices and edges respectively. Holonification is the problem of finding set of holons, H, such that:

$$H = \{h_1,h_2,...,h_k\}; h_i = \{a^i_1,a^i_2,...,a^i_{n_i}\} \mid \cup_i h_i = V \quad (1)$$

In Eq. 1, a^i_j denotes the j-th agent in holon i; n is the total number of agent in h_i and k is the number of holons and $h_i, i = 1,2,...,k$ are the holons. Generally, the holons may overlap, i.e. there are some members who belong to more than one holon; However, since many real world complex applications use non-overlapping communities as sub-structures, we assumed that the holons are disjoint, that is: $h_i \cap h_j = \emptyset; i \neq j$. We represent the concept of dependencies in multi-agent systems through the weights of the graph. Below, a detailed explanation of the dependency concept is given. It is also explained how the quality of holons is measured.

Dependency. In a multi-agent system, decision of an agent in the network influences some other agents. It depends on the state that whether an agent acts independently [14]. These influences are context-specific and can be qualified by a *dependency measure*. In our graph-based modelling, each dependency is considered as an edge in the graph and the degree of dependencies is defined by weight value. In literature there are some works pertaining to the learning of these dependencies. Among these works we can mention [14] and [15].

Since the definition of dependency usually varies in different applications and is based on the modelling purpose, we skip to discuss more about this concept and without loss of generality we assume that the dependency of two agents, a_i and a_j is available to the algorithm through a function, $d(a_i, a_j)$.

Quality Measure. Holonification reduces the complexity of the large networks by partitioning the agent's network. It divides the network to much simpler units such that the overall computational decreases, i.e. large scale problem is converted to multiple simple sub-problems and the computational load of the large network is scaled down. In our approach holonification tries to find a proper set of holons that maximizes a quality measure whose definition depends on the application. In this paper, it has been assumed that more dependent agents belong to same holons without any restriction on the number of holons and the number of agents in them. We define the quality measure as:

$$Q(H) = \sum_{h_i \in H} \frac{1}{n_i} \sum_{a^i_j, a^i_k \in h_i} d(a^i_j, a^i_k) \tag{2}$$

where $H = \{h_1, h_2, ..., h_k\}$ is the set of holons; n_i is the number of agents in h_i; a^i_j and a^i_k are two agents that are members of h_i; and $d(a^i_j, a^i_k)$ is the dependency function defined for two agents. As it can be seen from Eq. 2 the quality measure computes the sum of average dependencies over the number of agents inside the holons. If we show the total number of agents in the network by N; the number of agents in holon h_i by n_i; and the number of intra-holon connections by c_i; then minimum quality measure, $Q(H) = 0$, occurs when every agent in the network belongs to a separate disjoint holon, that is: $\forall h_i \in H; h_i = \{a^i_1\}$. In order to compute the maximum value of $Q(H)$, we have:

$$\sum_{h_i \in H} \frac{1}{n_i} \sum_{a^i_j, a^i_k \in h_i} d(a^i_j, a^i_k) \quad \leq \sum_{h_i \in H} \frac{1}{n_i} c_i \times \max_{j,k}[d(a^i_j, a^i_k)] \tag{3}$$

Considering the fact that maximum number of connection is obtained in a complete graph model which yields the maximum value for c_i as $\frac{n_i(n_i-1)}{2}$, we have:

$$\begin{aligned} Q(H) \leq \frac{1}{2} \sum_{h_i \in H} \max_{j,k} d(a^i_j, a^i_k) &\leq \frac{1}{2} \max_i [\max_{j,k}[d(a^i_j, a^i_k)]] \times \sum_{h_i \in H} (n_i - 1) \\ &= \frac{1}{2} \max_{i=1,\dots,k; h_i} (\max_{j,k}[d(a^i_j, a^i_k)] \times (N-k) \end{aligned} \tag{4}$$

According to Eq. 4 the minimum quality measure, $Q(H) = 0$, happens when $N = k$ and maximum quality measure happens when $k = 1$. It means that there is just one holon that all agents are the member of it. It definitely happens when all holons (sub-graph) are a complete graph. In this case, the output of the proposed holonification method will be one holon for such a network.

3.2 Holon Formation

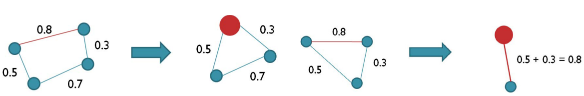
At the beginning of the holonification process, it is assumed that we have a network of agents modelled by $G =< V, E >$ which is a weighted undirected graph constructed according to the procedure described before. The algorithm starts with an empty set of holons and during the algorithm, new holons are added according to following steps: first, the process of choosing the best candidates to form a holon or to join an existing holon. Second, the reformation of the graph to reflect the changes made in first step.

As for the first sub-process, a greedy method is applied. In this method at every step an unvisited edge of the graph G is chosen in such a way that its weight is the maximum among all other unvisited edges. There are three possibilities: (i) the end nodes of the chosen edge do not belong to any existing holon; and (ii) one of the end nodes of the edge belongs to an existing holon and (iii) both end nodes of the edge belongs to an existing holon. Concerning the first case, a new holon is created. Since the overall measurement is always improved, this newly created holon is added to the list of the existing holons and adding the end nodes of the edge to the created holon, they are all marked as being visited.

When the second case occurs, it is assumed that the corresponding holon grows to hold the other end node of the edge. Now the measurement is calculated for the new holon considering its new structure. If it is improved, the holon is changed into its new structure. Finally in case of (iii) two holons corresponding to the two end nodes of the edge are merged. This way, a new and larger holon is created which contains all of the members of the merged holons. Then the measurement is calculated for the new holon. If it is improved, which will cause an improvement in the overall measurement, this new holon replaces with the last two small holons.

When an appropriate decision about the growth of the holons is made in every step, the structure of the graph (agents network) is reformed. In this reformation we attempt to treat the newly created holons as a new node in the graph and as the representative of its member nodes (agents). This is done in order to facilitate the decision making process. For the reformation of the graph we define two general rules as follows: (i) a node is created for each newly constructed holon. now all of the edges pointing to the members of the created holon, are corrected to point to the representative node. (ii) when an existing holon is grown, all of the edges pointing to its members point to the representative node of the grown holon.

In the process of reforming the graph and correction of the connections (edges), two cases occur: (i) there are no similar edges concerning their end nodes after the correction of the edges as described above. In this way the weights of the edges are not changed. An example is shown in Fig. 2(a). (ii) there are two or more similar edges corresponding their end nodes. In this case, a new edge represents the similar edges and its weight is the sum of the similar edges. Fig. 2(b) demonstrates this concept. These sets of steps are applied in every phase of the algorithm in which a new holon is identified or an existing one is grown. The holonification algorithm terminates when there is no unvisited edge left in the graph. At the end, for every singleton node that does not belong to any holon, a new holon is created with that node as its only member.



(a) There are no similar edges. (b) There are two or more similar edges.

Fig. 2. Two examples for holonification

It should be noted that, the original structure of the graph, concerning the actual edges (dependencies) between the nodes (agents) is not affected and changed by this algorithm. In other words all of the manipulations described above are applied temporarily to identify the proper holons in the graph. The holonification process is described in the algorithm 1.

```
Holon = ∅;
while there_is_an_unvisited_edge do
    e1 = edge_with_max_weight();
    if holonOf(end_nodes(e1)) = ∅ then
        h1 = newHolon(end_nodes(e1));
        add(Holon, h1);
        rep_node = newRepresentativeNode();
        reform_graph();
    end
    else if e1.onlyOneEndIsInHolon() then
        h1 = connectedHolon(e1);
        newQ = (h1.Q * h1.numOfNodes() + e1.weight)/(h1.numberOfNodes() + 1);
        if(newQ > h1.Q)
        h1.grow(e1);
        reform_graph();
    end
    else if e1.two_ends_are_in_holon() then
        h1 = first_connected_holon(e1);
        h2 = second_connected_holon(e2);
        newQ =
        (h1.Q * h1.numOfNodes() + h2.Q * h2.numberOfNodes() + e1.weight)/
        (h1.numOfNodes() + h2.numberOfNodes());
        if newQ > (h1.Q + h2.Q) then
            Merge(h1, h2);
            Reform_graph();
        end
    end
end
returnHolon;
```

Algorithm 1. The proposed holonification algorithm

4 Case Study: Urban Traffic Network

The problem of intelligent traffic control has been studied for many years. Agent-based urban traffic models have been found as efficient tools for traffic planning. The application of multi-agent systems to urban traffic control has made it possible to create and deploy more intelligent traffic management. As the complexity of traffic control on a network grows, it becomes more difficult to coordinate the actions of a large number of traffic entities that are available in the network. One way of handling this complexity is to divide the coordination problem into smaller coherent sub-problems that can be solved with a minimum number of interactions [16]. A common way in agent-based simulation of urban traffic is to consider each intersection as an agent. So the intersection map forms a network of interacting agents [17],[16], [18]. There are some researches that have used a graph model to represent the traffic network [19], [20]. The problems and simulations related to urban traffic networks can be a good test bed for holonic solutions due to its dynamic nature and possibly large set of entities. In this case study, we will see how our algorithm can be applied for the modelling and holonification of the network.

4.1 Holonification

We model the intersections as the nodes of the graph and the roads as the edges. The resulting graph has the following properties: *Connected*: there is at least one path between two intersections. *Planar* : It is assumed that the network has an intersection wherever two roads cross. *Weighted*: every two adjacent intersections are dependant physically considering their traffic flow. These dependencies are specified by the weights of the graph. Dependency between two nodes is measured by a weight value. *Undirected*: It has been assumed that the dependency is bidirectional.

Let's have a traffic network consisting of 14 intersections and 22 connecting roads between the intersections. Making a graph model of this network using the method described above, we will have a graph with $|V| = 14$ and $|E| = 22$ as shown in Fig. 3(a). The dependency between every two intersections is computed according to the distance between them and the traffic rate flowed to each other. It can be easily seen that there are many ways for putting the nodes (agents) inside of holons. Finding the best holonification that maximizes our quality measure is an NP-complete problem. For this example, we choose three adjacent nodes to make a holon, the resulting set

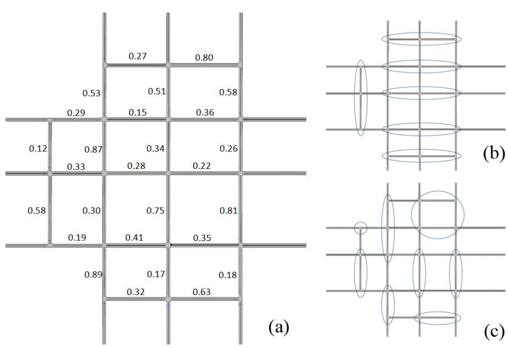


Fig. 3. (a)An example of traffic network modelled by a weighted graph; (b)a random homogeneous holonification; (c) The proposed holonification method

of holons is shown in Fig. 3(b) that the quality measure of it is 1.496 according to Eq. 2. The result of the proposed holonification method is depicted in Fig.3(c) which gives a quality measure of 2.859. As it can be seen, the proposed method has produced a more optimal holonification.

4.2 Intra-holon Timing

In this section, a simple method is utilized to assign green time to different phases of the junctions. Each agent placed at one junction and cooperates with the other connected agents belonging to the same holon by using the dynamic information provided by them. This information is the number of the cars passed through those junction in a specific time interval. The agent, then, uses this information together with the dependencies of its connected nodes to compute the green time of its phases. Formally, let's d_{ij} be the dependency between the agents i and j; T be the cycle time; N_j^t be the number of the cars passed through junction j at t-th time interval. Assigning a separate signal phase to each approaching link of a junction, the green time of the phases of the links outside of the holons are computed by Eq. 5 where i shows the current junction we are assigning the green times and GT_{ij}^t is the green time for the phases corresponding to the link connecting junctions i and j at t-th time interval.

$$GT_{ij}^t = \frac{d_{ij}}{\sum\limits_{k \in neighbor(i)} d_{ik}} \times T \tag{5}$$

. Similarly, for the green times of the phases of the links inside the holon, we have:

$$GT_{ij}^t = \frac{1}{\sum\limits_{k \in neighbor(i)} d_{ik}} \times T \times \frac{d_{ij} \times N_j^{t-1}}{\sum\limits_{l \in neighbor(i); l \in holon(i)} d_{il} \times N_l^{t-1}} \tag{6}$$

where $neighbor(i)$ denotes the set of agents adjacent to agent i and $holon(i)$ denotes the holon which i is a member.

4.3 Experimental Results

The proposed method has been tested on a network that contains 18 intersections and 39 two-way links using the Aimsun traffic simulator [1]. The network configuration parameters are shown in Table 1.

We performed some experiments to determine the average delay time of the network for different holonic structures. To show the impact of holonification, consider the signals timing method mentioned before. It has been tested on the network for two structures, a homogeneous method and the proposed method. The results show the proposed method has reduced the average delay time in comparison with the other one. Figure 4 shows the delay time over the mentioned traffic demand for two networks

[1] http://www.aimsun.com

Table 1. Network configuration parameters

Properties	Value	Properties	Value
number of intersections	18	number of lanes per links	3
number of links	78	arrival distribution	Uniform
average length of links	425.38m	simulation duration	10hour
maximum speed	60km/h	traffic demand	5000 veh/hour

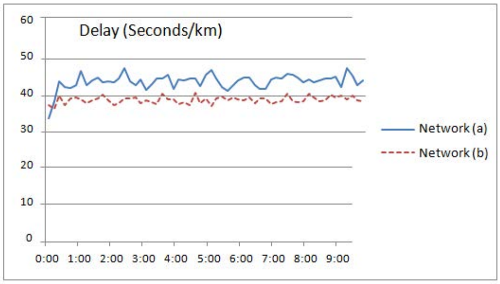


Fig. 4. Delay time tested on both networks

shown in Figs. 3(b) and 3(c). It can be seen that the delay time of the signal timing based on the proposed holonification is less than the one based on homogeneous method.

5 Conclusion and Future Works

In many real world applications we deal with systems that are complex and have a large scale. Holonic multi-agent system can be used as an effective tool for tackling these systems. Holonic systems need to specify the holons (the process of holonification). In this paper we have proposed a holonification method which can be applied to a population of inter-dependant agents in order to recognize and form the holons. In the proposed method, we modelled the agents using an undirected and weighted graph which the weights denote the degrees of the dependencies between the agents and a greedy algorithm was proposed to form the initial holons. The strength of the proposed method was demonstrated in an urban traffic network under two holonic organizations. Furthermore, a general quality measure based on the dependencies between the agents was introduced in this paper. Future works include working on the definition of quality measure based on the properties that are expected from the holons and extends the method for more levels and dynamic holarchies.

Acknowledgements. The first authors would like to thank Research Institute for Information and Communication Technology - ITRC (Tehran, Iran) for their supports.

References

1. Weiss, G.: Multiagent Systems: A Modern Approach to Distributed Modern Approach to Artificial Intelligence. MIT Press (1999)
2. Theocharopoulou, C., Partsakoulakis, I., Vouros, G.A., Stergiou, K.: Overlay networks for task allocation and coordination in dynamic large-scale networks of cooperative agents. In: Proc. of 6th International Joint Conference on Autonomous Agents and Multi Agent Systems, Hawaii, pp. 295–302 (2007)
3. Irandoust, H., Benaskeur, A.R.: Multi-organizational structures. In: Proc. of Association for the Advancement of Artificial Intelligence, Chicago, pp. 25–33 (2008)
4. Tianfield, H.: A new framework of holonic self-organization for multi-agent systems. In: Proc. of IEEE International Conference on Systems, Man, and Cybernetics, Montreal, pp. 753–758 (2007)
5. Koestler, A.: The Ghost in the Machine. Hutchinson (1967)
6. Rodriguez, S., Hilaire, V., Koukam, A.: Towards a holonic multiple aspect analysis and modeling approach for complex systems: Application to the simulation of industrial plants. Simulation Modelling Practice and Theory 15, 521–543 (2007)
7. Rodriguez, S., Hilaire, V., Koukam, A.: Formal specification of holonic multi-agent. In: Proc. of International Conference on Computational Science, pp. 719–726 (2005)
8. Deen, S.M.: A cooperation framework for holonic interactions in manufacturing. In: Proc. of the Second International Working Conference on CKBS, pp. 103–124 (1994)
9. Burckert, H.J., Fischer, K., Vierke, G.: Transportation scheduling with holonic mas - the teletruck approach. In: 3d International Conference on Practical Applications of Intelligent Agents and Multiagents, pp. 577–590 (1998)
10. Rodriguez, S., Hilaire, V., Koukam, A.: Towards a methodological framework for holonic multi-agent systems. In: Proc. of Fourth International Workshop of Engineering Societies in the Agents World, London, pp. 179–185 (2003)
11. Ulieru, M., Geras, A.: Emergent holarchies for e-health applications: a case in glaucoma diagnosis. In: Proc. of 28th Annual Conference of the Industrial Electronics Society, vol. 4, pp. 2957–2961 (2002)
12. Brandes, U., Delling, D., Gaertler, M., Gorke, R.: On finding graph clusterings with maximum modularity. In: Proc. of 33rd International Workshop on Graph-Theoric Concepts in Computer Science, pp. 121–132 (2007)
13. Schaeffer, S.E.: Survey graph clustering. Computer Science Review, 27–64 (2007)
14. Kok, J.R., Hoen, P.J., Bakker, B., Vlassis, N.: Utile coordination: Learning interdependencies among cooperative agents. In: Proc. of IEEE Symposium on Computational Intelligence and Games, Colchester, pp. 29–36 (2005)
15. Melo, F.S., Veloso, M.: Learning of coordination: Exploiting sparse interactions in multiagent systems. In: Proc. of 8th International Conference on Autonomous Agents and Multi Agent Systems, Budapest, pp. 773–780 (2009)
16. Doniec, A., Mandiau, R., Piechowiak, S., Espie, S.: A behavioral multi agent for road traffic simulation. Engineering Applications of Artificial Intelligence 21, 1443–1454 (2008)
17. Hirankitti, V., Krohkaew, J., Hogger, C.: A multi-agent approach for intelligent traffic-light control. In: World Congress on Engineering, London (2007)
18. Zhang, H.T., Yu, F., Li, W.: Step-coordination algorithm of traffic control based on multi-agent system. International Journal of Automation and Computing 6, 308–313 (2009)
19. Hu, M.B., Jiang, R., Wu, Y.H., Wang, W.X., Wu, Q.S.: Urban traffic from the perspective of dual graph. The European Physical Journal B(63), 127–133 (2008)
20. Chen, S.W., Yang, C.B., Peng, Y.H.: Algorithms for the traffic light setting problem on the graph model. In: Taiwanese Association for Artificial Intelligence, pp. 1–8 (2007)

Correct Speech Visemes as a Root of Total Communication Method for Deaf People

Eva Pajorová and Ladislav Hluchý

Institute of Informatic, Slovak Academy of Sciences,
Dúbravská 9 84507 Bratislava Slovakia
{Eva Pajorová,utrrepaj}@savba.sk

Abstract. Many deaf people are using lip reading as a main communication fiorm. A viseme is a representational unit used to classify speech sounds in the visual domain and describes the particular facial and oral positions and movements that occur alongside the voicing of phonemes. A design tool for creating correct speech visemes is designed. It's composed of 5 modules; one module for creating phonemes, one module for creating 3D speech visemes, one module for facial expression and modul for synchronization between phonemes and visemes and lastly one module to generate speech triphones. We are testing the correctness of generated visemes on Slovak speech domains. The paper descriebes our developed tool.

1 Introduction

The reason to develop our tool comes from the requirements of deaf people. .Analysis of the communication used by deaf people shows that 80% of deaf people have interest to understand normal speech [10]. The deaf community want to interact freely with normal speeking people. They prefer oral communication before sign language. Over the years there have been many debates and studies done on how to communicate and educate to/with the deaf and hard-of-hearing child. At first, people believed that deaf children were incapable of learning and so didn't bother trying to communicate or educate them. But, over the years it was proved that deaf children were capable and could learn to communicate and wanted to communicate, just like other people. Today, the issue has become; what is the best way to try and educate and communicate to/with/for the deaf [12], [13] ?

There are three main methods that have been developed and are used:

- the Oral method (oralism)
- the Manual method (manualism)
- the Total Communication method

Each of these methods had various pros and cons; all should be carefully examined by the parents of a deaf child. The ultimate method for communication should be chosen based on how the child can be empowered and function in society, sadly hearing people don't always consider what is best for the Deaf.

The Oral method is a method of communication and educating for deaf and hard-ofhearing children using only the spoken language, lip reading, and voice training.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 389–395, 2012.

The The manual method or manualism is based totally on Sign language and using the hands or other physical ways to communicate. The goal of this method is to provide a way for deaf people to interact with others without the use of spoken language. Children taught using this method don't need to worry about learning to speak or develop lip reading skills. This method is considered the natural way that deaf children learn to communicate. Furthermore, it encourages a sense of pride in being Deaf.

Total Communication is a fairly new method for educating and communicating with deaf and hard-of-hearing children. The goal of this method is to incorporate lip-reading, speech, and sign language so that a child can communicate effectively in almost any setting. Children taught using this method are given an opportunity to develop their voice as much as possible, as well as allowed to use the more natural, manual/visual way of communicating. The results are amazing, since children are taught how to interact in both the hearing and the deaf world. Total Communication works well to educate deaf and hard-of-hearing children, since it doesn't matter if a child has a mild hearing loss or is profoundly deaf.This method works with the child, the parents, and the educators, so that everyone can learn and communicate to the best of their ability. Children are allowed to be themselves and learn in a way that suits their needs. Total Communication includes everyone -- the hearing world and the deaf world.

So, what is the best way to try and educate and communicate with/for the deaf, oralism, manualism, or Total Communication?

After analyse we prefer Total Communication which to incorporate lip-reading, speech and sign language and it means that the deaf child can communicate effectively.

The main goal of our work is to develop correct visemes for lip-reading. We test the results on virtual slovak speaking head. We are using the triphone based approach.

For generating of new sentences, we use a triphonebased approach [6]. Triphones are short pieces of motion sequence that span three phonemes, so each viseme is stored with its context and therefore captures all of the coarticulation effect caused by the direct neighbors. Our similarity measure is easily extended from visemes to triphones, and we can thus find the best overlapping triphone sequences in our database that match any new sentences that needs to be synthesized. Our work is based on dense 3D surface scans, which makes it more versatile than image-based techniques [6].

Related Work Facial animation is facing three different challenges:

1. producing corect and realistic face shapes in every single frame of the animation
2. creating a dynamically realistic face motion over time
3. creating corect and realistic lip-speech animation

Many models [4,5,6,7] are based on marker point positions , 3D scans or images. These approaches face the problem of defining how the parameters of the model vary over time. For speech synthesis, this involves the problem of coarticulation. Consecutive new approach: "Modeling coarticulation in synthetic visual speech" [8] define dominance functions of phonemes that control the interaction between subsequent phonemes as applied to muscle-based systems [1]. The systems based on Hidden Markov Model [9] to learn the dynamics of speech from audio, and transfer this information to a face model. Another approach [10]uses regularization techniques to compute smooth curves for the model parameters over time. In this model, coarticulation is due to the smoothness of the curve and a statistical representation of the variance of each viseme, instead of synthesizing motion entirely. New approach which bring novelty was build on Video Rewrite[11].

Our goal is use the last one sophisticated Video Rewrite approach and arrange the best work bench for developing real speaking head in Slovak language. Video Rewrite method deduce a phoneme similarity measure. This quantitative similarity measure relaxes the selection rule of viseme grouping and offers further substitution options. Similarly to Video Rewrite, the optimal triphone sequence for the synthetic animation is found by minimizing an error function that takes both, viseme similarity and smoothness of the animation into account.

In our work, we are using the software system: "Lipsynctool": for a statistical analysis of the distribution of mouth configurations in high-dimensional face space. The Lipsynctool is an interactive application that allows users to create lipsync animation from audio files. The Lipsynctool is based on laser light engines (LLE). This method estimates a lowdimensional, nonlinear manifold from a set of data points. In the Lipsynctool system, LLE allows us to derive a highly specific criterion for viseme similarity that dictates appropriate triphone substitutions. LLE has been used previously by Wang et al.[28] as a representation that allows the separation of expression style from expression content with a bilinear model. Using a closely related Isomap method, Deng and Neumann[11] present a data-driven approach to speech animation where users can edit facial expressions in sequences.

In our work, we proposed a new system to automatic Slovak speech animation. Our novel selection method takes full advantage of the dual association between phonemes and visemes: not only can a phoneme take the visual appearance of several visemes, but visemes can be attributed to different phonemes as well. Our method determines visemes that can be used as a valid substitution for a specific phoneme event.

2 Tool for Creating Slovak Speech Visemes

The developed tool for creating Slovak speech visemes is composed of 5 modules. One modul for creating phonemes, one modul for creating 3D slovak speech visemes, one modul for facial expression and one modul for synchronization between phonemes and visemes and lastly one modut to generate slovak speech triphones (Fig. 1. is a schema of the momules involved in the developed tool). Each of the modules can operate as a separate module or can be included to the framework. For the synchronization of all modules we use Lipsynctool.

Lipsynctool is able to synchronise phonemes together with creating own character visemes and with facial expresion module. Es a final triphones are developed with triphone mudules which are animate on the self developed face models.

All modules are creating tool for developing Slovak speech speaking heat.Virtual slovak speaking head is able navigate and instruct the crisis situations as is evacuation the people from big halls, also to evacuation shopping centre and its could be useful in crisis situation as are natural disasters (fires floods...).

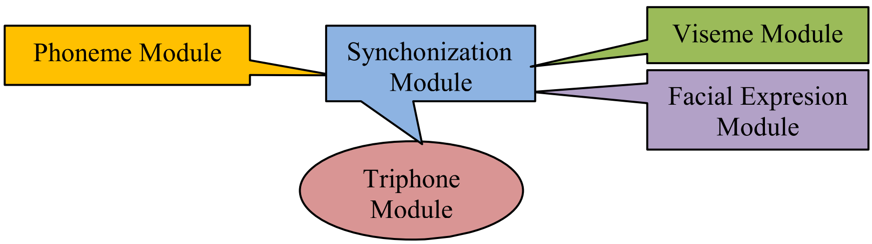


Fig. 1. Schema of modules of tool for Slovak speech visemes

2.1 Phoneme Module

For Slovak speech phonemes input we use phonemes which have been generated by Slovak speech synthesiser - developed in our institute [1],[2],[3]. In a scope of the activities was developed lot of Slovak speech results as is for example the text collection Slovak National Corpus. Other activity is focused on speech database building for speech recognition purposes. It was developed and build a one speaker speech database in Slovak for experiments and application building in unit-selection speech synthesis. To build such a database, we exploit as much of the existing speech resources in Slovak as possible, and utilize the knowledge from previous projects and the existing routines developed at our department [12]. As an input file, an audio file (.wav) and Slovak text was used (see Fig. 2.).

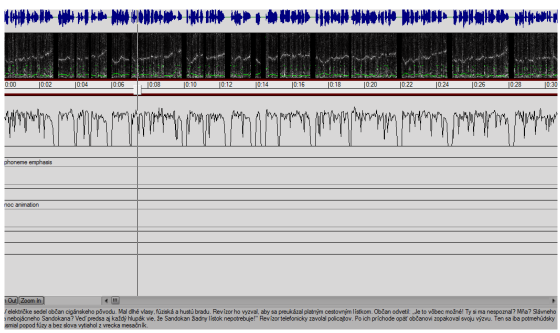


Fig. 2. Slovak speech phonemes from audio file

2.2 Viseme Module

The viseme targets for our character should be somewhat exaggerated from real mouth positions. We use frames from a laser scanner (see Fig. 3) for design of the visemes. The mouth positions can be approximately grouped into vowels and consonants. Vowels are the voiced sounds. At their peak position, they will have a comparatively open position to its neighboring consonants.

Realism is affected by three factors in the viseme representation:

- *The width/backness* of the mouth.
- *The openness of the mouth.*
- *The rounded-forwardness of the mouth.*

The speech functionality generates articulation information that controls the openness and emphasis of a phoneme per frame (phn_vis) or on a curve (phn_env). Mouths should be created as if they were being emphasized. The speech system will generate information to control "how much" of the phoneme is turned on. The speech system won't exaggerate the phoneme unless the speech is emphasized.That said, certain phonemes are more open than others by default.

For good lipsync, it is necessary to try to capture the various curved phonemes. Each phoneme will be a combination of those attributes. Most are a combination of 2, openness and rounded-forwardness.

Referring the to viseme images, we can classify each phoneme as the contribution of a width-backness factor, the openness factor, and the rounded-forward factor. This is subjective, and there may be disagreement. The viseme is listed following by a set of contributions. If the contribution is zero, this means the "neutral position" for forward-rounded, width-backness. For openness, zero means closed.

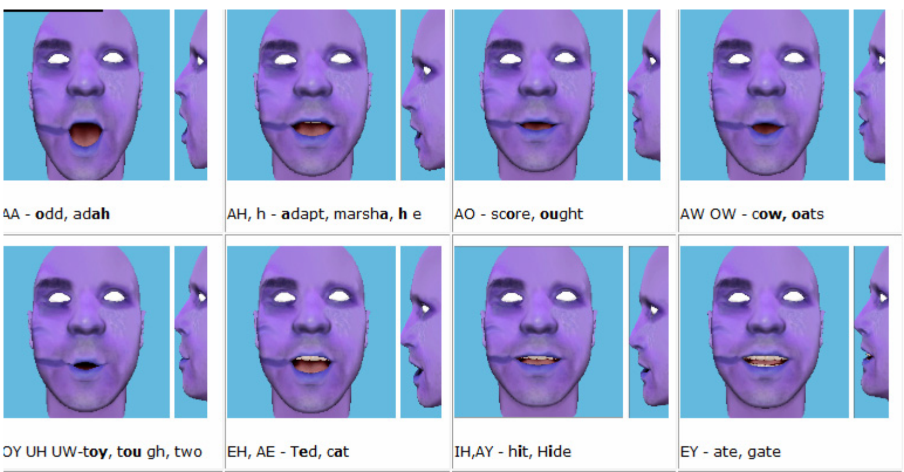


Fig. 3. Viseme examples

2.3 Module for Facial Expression

Expression is a group of facial parameter values that together transform the neutral face into an expressive face [2]. Expressions simulation and animation can be divided into two groups.The first one is animated or recorded visemes (and corresponding phonemes), which act on the region of the lips and the mouth and form segments of words. The second one is emotions which act on any part of the face. Contrary to other regions of the human face, the lips are mainly characterized by their contours. Several models of lips were created. One of them uses algebraic equations for contours approximation. From a multidimensional analysis of a real speaker's gestures, visemes and corresponding phonemes can be extracted. After the lips, the most visible moving part of the face due to speaking is the jaw. Jaw kinematics can be modeled manually or automatically from the data recorded by video, mechanical or opto-electronic sensors. In realistic image synthesis, superpositions of jaw transformations and lips shapes must be integrated with emotions. Primary emotions are for example surprise, fear, anger, happiness and sadness. Basic expressions and their variants can be defined using snapshots from the computer or real model of the actor [3].

2.4 Synchronization Module

Phonemes are expressed using a combination of jaw, tongue, and lip movements. Each of these parts of the mouth is called a channel in the Speech action clip. Having all three channel animations within one clip helps you create smooth blending between the visemes because you can control the lip-sync animation as one entity. Within the clip, however, the animation of the lip, jaw, and tongue controls are separated into different curves so that you can control each channel on its own. Annosoft Lipsync automatically determines how open the mouth should be, and when. The viseme should try to be as accurate a representation as possible, obviously. To get there we need to subdivide the vowels so that we can make an accurate mouth reference.

2.5 Triphone Module

In linguistics, a triphone is a sequence of three phonemes. Triphones are useful in models of natural language processing where they are used to establish the various contexts in which a phoneme can occur in a particular . The Lipsynctool allows us to define an arbitrary number of visemes. The Lipsynctool is an application providing powerful automatic lip-sync synchronization and timeline editing capabilities

3 Conclusion

The goal of our work is to develop such sophisticated tool, which will be provide the regular Slovak speaking head. We are testing the Slovak visemes in a Lip-reading method for Slovak deaf people. Currently we try to include them to the Total Communication method. Lip-reading together with sing language could be help the Slovak deaf people. The most popular systems used to transfer visual information for distance training for hearing-impaired people via Internet, which is based on the video compression standards H.263 and MPEG-4. In this case, the basic information is represented with video sequences, containing the image of the Sign Language interpreter [1].

Acknowledgments. „This contribution/publication is the result of the project implementation:Research and development of new information technologies for forecasting and solving crisis situations and for the security of populace, ITMS code: 26240220060 supported by Operational Programme Research & Development funded by the ERDF.

References

1. Hrúz, M., Krňoul, Z., Campr, P., Muller Ludek, M.S.: Tovards Automatic Annotation of Sing Language Dictionarz Corpora. In: TDS Pilsen, pp. 331–339 (2011)
2. Cai, Q., Gallup, D., Zhang, C., Zhang, Z.: 3D deformable face tracking with a commodity depth camera. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part III. LNCS, vol. 6313, pp. 229–242. Springer, Heidelberg (2010)
3. Drahoš, P., Šperka, M.: Face Expressions Animation in e-Learning, `http://lconf06.dei.uc.pt/pdfs/paper14.pdf`
4. Albrecht, J.: Haber, and H.-P. Seidel. Speech Synchronization for Physicsbased Facial Animation. In: Skala, V. (ed.) Proc. 10th Int. Conf. on Computer Graphics, Visualization and Computer Vision (WSCG 2002), pp. 9–16 (2002)
5. Ma, J., Cole, R., Pellom, B., Ward, W., Wise, B.: Accurate visible speech synthesis based on concatenating variable length motion capture data. IEEE
6. Wang, A., Emmi, M., Faloutsos, P.: Assembling an expressive facial animation system. In: Sandbox 2007: Proceedings of the 2007 ACM SIGGRAPH Symposium on Video Games, pp. 21–26. ACM Press (2007)
7. Ezzat, T., Geiger, G., Poggio, T.: Trainable videorealistic speech animation. In: SIGGRAPH 2002: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, pp. 388–398. ACM Press (2002)
8. Cohen, M.M., Massaro, D.W.: Modeling coarticulation in synthetic visual speech. In: Magnenat Thalmann, N., Thalmann, D. (eds.) Models and Techniques in Computer Animation, pp. 139–156. Springer, Tokyo (1994)
9. Brand, M.: Voice puppetry. In: SIGGRAPH 1999: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, pp. 21–28. ACM Press/Addison-Wesley Publishing Co. (1999)
10. Kim, I.-J., Ko., H.-S.: 3d lip-synch generation with data-faithful machine learning. Computer Graphics Forum. In: EUROGRAPHICS 2007, vol. 26(3) (2007)
11. Bregler, C., Covell, M., Slaney, M.: Video rewrite:driving visual speech with audio. In: Computer Graphics, Proc. SIGGRAPH 1997, pp. 67–74 (1997)
12. Meadow-Orlans, K.P., Mertens, D.M., Marilyn, S.-L.: Parents and their Deaf Children. Gallaudet University Press, Washington D.C. (2003)
13. Munoz-Baell, I.M., Ruiz, T.M.: Empowering the deaf; let the deaf be deaf. Journal Epidemiol Community Health 54(1), 40–44 (2000)

Target Depth and Metal Type Discrimination with Cost Based Probability Updating

Minh Dao-Johnson Tran[1], Canicious Abeynayake[2], and Lakhmi C. Jain[1]

[1] Knowledge-Based Intelligent Engineering Systems (KES) Centre,
School of Electrical and Information Engineering,
University of South Australia, Australia
tramd005@mymail.unisa.edu.au

[2] Threat Mitigation Group, Weapon Systems Division,
Defence Science and Technology Organisation (DSTO), Australia
Canicious.Abeynayake@dsto.defence.gov.au

Abstract. This paper presents a cost based probability updating technique based on Bayesian decision theory for modifying decision confidences. The technique allows for the incorporation of prior knowledge via the use of cost matrices supplied by another source of intelligence. Data signals acquired by a metal detector array corresponding to UXO based targets were used to evaluate the technique with the assistance of a previously developed automated decision system. The classification classes utilised were based on target depth level and metal type. The results showed that the probability updating technique was able to produce an increase in the classification performance and also reduce the classification errors below approximately 5 to 10%.

Keywords: Bayesian cost probability, target property discrimination, metal detector, automated decision system, UXO.

1 Introduction

Military, industrial, and improvised explosive devices (IEDs) are a significant problem in many countries around the world. Examples of these threats include landmines and unexploded ordnance (UXO) used and abandoned both during and after periods of conflict. A number of different technologies have been trialled and operated to detect these explosive threats such as ground penetrating and synthetic aperture radar (GPR, SAR) [1, 2], biological sensors (e.g. canines, rats, gerbils) [3–5], infrared (IR) [6], and acoustic devices [2]. However, the most common and actively used technology in the field to date is the metal detector (MD) [7].

MD sensors are used to detect the metallic components found in the composition of many explosive devices. However, the sensor itself has no inherent means of discriminating between targets of interest and irrelevant metallic clutter. These clutter objects are a common occurrence in many areas where the threats are located, and are the source of a significant number of false alarms.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 396–405, 2012.

In the past human operators have been utilised to perform the discrimination process but this was often laborious, time consuming, and sometimes placed the wellbeing of the operators at risk. More recently software based approaches have been implemented for this task, however these systems are still limited and prone to high false alarm rates (FARs). A number of these systems also still require a heavy reliance on human operators to perform the discrimination despite the use of software tools.

This paper presents a cost based probability updating technique for modifying decision confidences, which allows for the incorporation of prior knowledge from another source of intelligence. The technique was implemented as part of an automated decision system network that was simulated using MATLAB. The data used to evaluate the system was collected over soil conditions considered similar to those found in operationally relevant scenarios. The target set used involved a number of high metallic UXO based targets buried at depths between flush and 48 inches. The MD sensor used to acquire the data was a Minelab Single Transmit Multiple Receive (STMR) Mark II (MkII) MD array, which consists of a single transmit coil and a set of ten dipole ('mono-loop') and quadrupole ('figure-8') receive coils. The receive coils are arranged in dipole-quadrupole pairs with each coil returning four signal channels labelled S, C, G and X. In this work only the signals returned by the dipole coils were analysed with each signal channel treated as a separate input stream.

The paper is arranged as follows; Section 2 briefly describes the current status of the discrimination problem, and the limitations of past and present techniques. Section 3 outlines the proposed technique and details its implementation in the automated decision system network. Section 4 presents and discusses the classification results obtained by the decision system network prior to and after applying the proposed technique. Finally section 5 summarises the paper and outlines directions for future research.

2 Current Status of the Discrimination Problem

Target discrimination of landmines and IEDs using MD technologies initially involved only distinguishing between possible target signals and background noise. Although high probability of detection (PD) is achievable at this level, pure detection from background noise often results in significantly high FARs due to the presence of benign metallic clutter. To address this issue, researchers began to develop techniques aimed at discriminating targets from benign metallic clutter. Despite the increase in the level of discrimination, a number of clutter items still bypassed the recognition process. To further reduce the FAR, research was conducted in differentiating between different types of targets as well as against clutter. Although this level of discrimination was achieved in a number of studies, the identification process was often adversely affect by the object's burial depth and orientation. Due to this dilemma, techniques have been investigated to infer specific information related to the detected targets. The variety of target information determined include properties such as its burial depth, size, and orientation.

For burial depth estimation a number of techniques have been proposed such as utilising the Finite Difference Time Domain (FDTD) technique combined with a radial-basis function neural network [8], the Standardised Excitation Approach (SEA) forward model coupled with a Support Vector Machine (SVM) [9], frequency analysis of scalograms generated using Gabor wavelets [10], basis function modelling combined with the subspace projection method [11], and analysing the ratio of field strengths from different vertical gradient MD coils using linear estimation and the quadratic polynomial discriminant (QPD) function [12]. Additional depth estimation techniques include analysing the ratio of the responses from two different sized or types of MD coils [13, 14], examining the ratio of energy from higher and lower frequency responses [13], and comparing the signal responses for corresponding data segments across different sensors [15].

In relation to the determination of other target properties, [16] presents a method for estimating an object's equivalent metal mass that involves analysing the object's electromagnetic induction (EMI) response in the form of an induced current. [17] presents a patent for determining a target's approximate size and shape in addition to its depth by analysing the signal response ratio between two different coil configurations. A similar patent is presented in [18] using the in-phase and quadrature components of the MD signal. In [19] a set of techniques are proposed for determining specific target properties. For burial depth and metal content discrimination, the sensor measurements from two different heights are analysed to exploit the distinct behaviour generated by targets buried at different depths with different levels of metal content. For target type classification, the length of the detection response above various thresholds is used as a feature for discrimination. Lastly for metal type classification, the phase for the in-phase and quadrature signals returned by the MD are calculated and used for discrimination.

Despite the number of techniques proposed, the overall research in the area of target property discrimination is still limited. The majority of the results presented in these studies are also still preliminary. In addition, one of the major limitations of past and current research is related to the evaluation data set. In a number of cases the data used to validate the technique is either synthetically generated, or taken under certain specific and often 'ideal' environmental conditions. As a result, these data sets are not realistic and do not represent the data actually found in operationally relevant scenarios. Another major limitation involves an assumption made in regards to the targets. Many techniques generally rely on the targets being purely homogeneous in composition (i.e. the target material exhibits either distinct magnetic or distinct conductive characteristics) [20, 21]. Due to this assumption, a number of the evaluation data sets have been generated using targets of pure metallic composition (i.e. pure copper, pure steel, etc.). However, this is not the case with targets encountered in realistic environments that are often composed of both magnetic and conductive components of arbitrary quantities. As a result, utilising these particular techniques generally becomes overly complex and/or infeasible.

3 Proposed Technique and Implementation

3.1 Automated Decision System

The automated decision system architecture utilised in this research was developed and evaluated in previous work [22–24]. The decision system is composed of three major components which are alarm generation, feature extraction and classification. Figure 1 is a schematic of the decision system architecture for a single input stream.

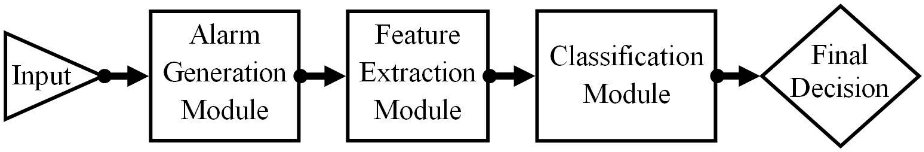


Fig. 1. Decision system architecture for a single input stream

The alarm generation component is designed to identify and isolate suspected areas in the input stream that may correspond to possible target signatures. This component was implemented using the peak detection technique described in [22, 23]. The feature extraction component processes the alarms to generate a discrete number of elements or 'feature vector' to uniquely represent the possible target signatures. The feature vectors used were composed of a set of wavelet and morphological based features generated and extracted from the MD signatures. Lastly the classification component analyses the feature vectors and determines the class they most likely belong to. This component was realised with a Fuzzy ARTMAP (FAM) neural network and used classification categories based on target depth level and metal type. The FAM network was also modified to return all possible classes/decisions together with their associated choice parameters, which also served as confidence values. The class with the greatest confidence value was then chosen as the final decision.

Since the MD channels were treated as a separate input streams, independent decision systems were used to process each channel. This produced a local decision for each channel, which were then synthesis using a decision fusion stage. The foundation of the decision fusion stage is based on the majority voting method, and operates by selecting the class with the greatest number of occurrences amongst the set of local decisions. The final confidence value is then computed as the mean of the confidence values from each occurrence. If no unique decision can be determined based on the number of occurrences, the decision with the greatest confidence value amongst all classes is chosen as the final decision.

As mentioned in section 2, many of the methods proposed in past and current research rely on the targets being purely homogeneous in composition. This is often not the case in realistic environments, which generally causes these techniques to become overly complex and/or infeasible to use. The decision system architecture utilised in this work does not suffer from this limitation as it does not rely on any assumptions made regarding the target's physical composition. As such, the system can readily adapt to variations in the target material and its surrounding environmental conditions unlike for example physics-based modelling techniques. Additionally since feature vectors are used to represent the target responses, the methodology can operate with any form of input regardless of the nature of the data provided meaningful features can be extracted.

3.2 Cost Based Probability Updating Technique

To incorporate prior knowledge into the decision making process a cost based probability updating technique was developed. This technique was applied to the classification results from each individual decision system prior to selecting the final local decision. In the context of the decision system operation, the technique allows the operator to either add a bias to the classification results corresponding to available target information, or orientate the decision system towards a particular set of interested targets.

The probability updating technique utilises the Bayesian conditional risk equation and considers only two possible hypotheses occurring during classification. The first hypothesis is the case when an arbitrary 'class x' is the true state (i.e. H_1). The second hypothesis is the contrary when class x is not the true state or 'not class x' (i.e. H_0). By applying the two category classification situation and a number of mathematical assumptions, it can be shown that the conditional risk equation can be rearranged and minimised to form the likelihood ratio

$$\frac{p\left(y|H_1\right)}{p\left(y|H_0\right)} = \frac{P_1\left(C_{10} - C_{00}\right)}{P_0\left(C_{01} - C_{11}\right)} \tag{1}$$

where P_a is the a priori probability of H_a (i.e. $P(H_a)$), $p(y|H_a)$ is the a posteriori probability of H_a with respect to y, and C_{ab} is the expected loss incurred if class a is selected given that the true class b is present. By treating the decision confidence value as an a priori probability, Equation 1 can be used to update the decision confidence by computing the a posteriori probability $p(y|H_1)$, which in turn becomes the updated confidence value. The operator can also incorporate their knowledge into the decision making process by manually modifying the cost variables C_{ab}. Therefore by adjusting the cost variables for particular class pairs, the operator can apply a bias towards a specific set of targets to modify their likelihood of being selected as the classification decision. In other words, the cost variables can increase (or decrease) the confidence value and hence the probability of selection for a particular set of targets according to the operator's preferences. This is especially beneficial in cases where the classification decisions corresponding to these targets originally may not have had the highest (or lowest) confidence values.

Since the a posteriori probability $p(y|H_1)$ is the only variable of concern, by exploiting a number of probability relations Equation 1 can be rewritten in terms of the corresponding a priori probability P_1 and cost variables C_{ab} to produce

$$p\left(y|H_1\right) = \frac{P_1 C}{1 + P_1\left(C - 1\right)} \tag{2}$$

where

$$C = \frac{C_{10} - C_{00}}{C_{01} - C_{11}} \tag{3}$$

The cost variables are supplied in the form 'cost matrices' as shown in Table 1. Every row and column in the cost matrix corresponds to one particular classification class. As a result, each cell value represents the cost variable C_{ij} where i is the matrix row and j is the matrix column (i.e. the cost of classifying class i as class j). As the cost variables are operator defined, the cost matrix entries were determined manually based on the number of individual target property samples available for classification. Note that for the cases when i is equal to j the cost variable is zero (i.e. no cost is incurred when class i is classified as itself). The cost matrices used in this study can be found listed in [23]. During the probability update calculations, cost variables C_{01} and C_{10} are determined by selecting the maximum and minimum available cost variables from the corresponding row and column (row for C_{10}, column for C_{01}). Two update calculations are performed using first the maximum selected cost variables, and then the minimum selected cost variables. The values are then compared and the greatest a posteriori probability of the two results is taken as the updated confidence value.

Table 1. Example cost matrix

Classification Type	**Class 1**	**Class 2**	**Class 3**	$\cdots$	**Class n**
Class 1	0	0.5	0.1	$\cdots$	0.8
Class 2	0.3	0	0.2	$\cdots$	0.1
Class 3	0.9	0.6	0	$\cdots$	0.5
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Class n	0.7	0.2	0.4	$\cdots$	0

4 Results and Discussion

Table 2 displays the performance achieved by each individual signal channel and after decision fusion for each target property classification. Table 2 shows the majority of the channels were able to archive classification performances of above 95% prior to applying probability updating. This was also achieved for both depth level and metal type classification, as well as after decision fusion was applied. The high performances may be attributed to the nature of the targets, which returned greater signal magnitudes due to their high metallic content. This resulted in more defined MD responses that in turn produced higher quality features. After probability updating was applied, an increase in the classification performance was observed across most of the results, while those that did not improve maintained their performance levels. Overall, the results for depth level classification were greater than the results for metal type classification.

Table 2. Classification performance before and after cost based probability updating

Target Class		Depth Level					Metal Type				
Channel		C	G	S	X	Decision Fusion	C	G	S	X	Decision Fusion
Probability Updating	Before (%)	98.32	98.32	98.74	95.38	98.32	96.64	96.64	94.12	78.99	94.12
	After (%)	99.58	99.58	98.74	99.58	100	96.64	98.32	94.12	79.83	97.06

Since perfect classification results were not achieved, confusion matrices were generated to better understand the cause of the errors. Table 3 and Table 4 are the dipole channel G confusion matrices based on the performance for depth level and metal type classification respectively. The depth levels are defined as SHALLOW (flush $\leq$ depth $\leq$ 5 cm), INTERMEDIATE (5 cm $<$ depth $\leq$ 30 cm) and DEEP (depth $>$ 30 cm). The metal types are defined by metal category, which in this case is only STEEL. However, due to the different variants of steel present in different targets, each steel variant is identified with a version number (i.e. STEEL.#01-03). The percentages in the confusion matrices are colour coded as follows; green represents correct classifications, cyan represents depth level only errors, and yellow represents metal type only errors. Although only the channel G confusion matrices are shown, the analysis described for this subset also approximately applies to the results for the remaining channels.

Table 3 shows that all depth level classes were able to achieved correct classifications above 95%. Alternatively, Table 4 shows that all metal type classes were also able to achieved correct classifications above 95% except for the class ‘STEEL.#03’. However, closer inspection revealed that all of the incorrect samples were misclassified as ‘STEEL.#02’. This may be due to the MD responses and hence the generated feature vectors of both target classes being similar. The reason for this may be that the STEEL.#03 metal type has a comparable physical composition to the STEEL.#02 metal type. However, further investigation is required to verify this hypothesis. Despite this, the STEEL.#03 class still achieved correct classifications of 90%.

Table 3. Dipole channel G confusion matrix for depth level classification performance

Depth Level		**Actual Class**		
		DEEP	INTERMEDIATE	SHALLOW
Predicted Class	DEEP	100	3.45	0
	INTERMEDIATE	0	96.55	0
	SHALLOW	0	0	100
	UNKNOWN	0	0	0

Table 4. Dipole channel G confusion matrix for metal type classification performance

Metal Type		**Actual Class**		
		STEEL.#01	STEEL.#02	STEEL.#03
Predicted Class	STEEL.#01	97.70	0.71	0
	STEEL.#02	2.30	99.29	10
	STEEL.#03	0	0	90
	UNKNOWN	0	0	0

5 Conclusion and Future Research

In this study, a cost based probability updating technique based on Bayesian decision theory for modifying decision confidences is presented. The probability updating technique allows for the incorporation of prior knowledge via the use of cost matrices supplied by the either the operator or another source of intelligence. The technique was evaluated using data signals acquired by a Minelab STMR MkII MD array corresponding to UXO based targets buried at various depths. The data signals were first processed using a previously developed automated decision system network prior to applying the updating technique.

The results showed that the probability updating technique was able to produce an increase in the classification performance across the majority of the channels and after decision fusion was applied. The classification error analysis also showed that the misclassifications generally resided below 5%. These observations were valid for both depth level and metal type classification. One metal type class did suffer from misclassifications above 5% in relation to another particular class. However, this was still only equivalent to 10% and was possibly attributed to similarities in the physical compositions of the corresponding targets.

Despite the promising results the proposed technique can still be further improved particularly in regards to the selection of the cost values. A more sophisticated method could be implemented for determining the initial cost values as well modifying them via incoming prior knowledge. This would allow the probability updating technique to be adaptable for real time applications.

Acknowledgements. Gratitude goes to the Threat Mitigation Group from the Defence Science and Technology Organisation (DSTO) for sharing the metal detector data used in this study.

References

1. Carin, L.: Ground-penetrating radar (paper i). In: Alternatives for Landmine Detection. RAND (2003)
2. Waschl, J.A.: A review of landmine detection. Technical Report DSTO-TR-0113, Defence Science and Technology Organisation: Aeronautical and Maritime Research Laboratory. Explosives Ordnance Division (December 1994)
3. Krausa, M., Massong, H., Rabenecker, P., Ziegler, H.: Chemical methods for the detection of mines and explosives. In: Detection of Explosives and Landmines: Methods and Field Experience. NATO science series. II, Mathematics, physics, and chemistry, vol. 66. Kluwer Academic Publishers, Dordrecht (2002)
4. Yinon, J.: Forensic and environmental detection of explosives. Wiley, Chichester (1999)
5. Harper, R.J., Furton, K.G.: Chapter 13: Biological detection of explosives. In: Yinon, J. (ed.) Counterterrorist Detection Techniques of Explosives, pp. 395–431. Elsevier B.V., Amsterdam (2007)
6. Baertlein, B.: Infrared/hyperspectral methods (paper i). In: Alternatives for Landmine Detection. RAND (2003)
7. Bruschini, C.: A Multidisciplinary Analysis of Frequency Domain Metal Detectors for Humanitarian Demining. PhD thesis, Vrije Universiteit Brussel (2002)
8. Zainud-Deen, S.H., El-Hadad, E.S., Awadalla, K.H.: Landmines detection using finite-difference time-domain and artificial neural networks. In: 14th International Symposium on Antenna Technology and Applied Electromagnetics & the American Electromagnetics Conference (ANTEM-AMEREM), pp. 1–4 (2010)
9. Fernandez, J.P., Sun, K., Barrowes, B., O'Neill, K., Shamatava, I., Shubitidze, F., Paulsen, K.D.: Inferring the location of buried uxo using a support vector machine. In: Detection and Remediation Technologies for Mines and Minelike Targets XII, vol. 6553, pp. 65530B–9. SPIE, Orlando (2007)
10. Trevelyan, J.: Target depth estimation for a metal detector in the frequency domain. In: Second International Conference on the Detection of Abandoned Land Mines (Conf. Publ. No. 458), pp. 218–221 (1998)
11. Marble, J.A., Yagle, A.E., Wakefield, G.H.: Physics derived basis pursuit in buried object identification using emi sensors. In: Detection and Remediation Technologies for Mines and Minelike Targets X, vol. 5794, pp. 1151–1159. SPIE, Orlando (2005)
12. Marble, J., McMichael, I., Reidy, D.: Estimating object depth using a vertical gradient metal detector. In: Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XIII, vol. 6953, pp. 695313–9. SPIE, Orlando (2008)

13. Marble, J., McMichael, I.: Metal detector depth estimation algorithms. In: Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XIV, vol. 7303, pp. 73030L–10. SPIE, Orlando (2009)
14. Abeynayake, C., Tran, M.D.J., Ferguson, M.: Investigation on improved target detection capabilities using advanced metal detector arrays. In: Land Warfare Conference 2010, Brisbane, Australia (2010)
15. Marble, J.A., McMichael, I.: Dual-emi system for object classification. In: Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XV, vol. 7664, pp. 76640B–8. SPIE, Orlando (2010)
16. Yan, X., Li, P., Liu, Z., Xu, F.: Metal equivalent measurement based on low-frequency electromagnetic induction method. In: IEEE Circuits and Systems International Conference on Testing and Diagnosis 2009 (ICTD 2009), pp. 1–4 (2009)
17. Mitchell, L.W.: Metal detector with improved detection capabilities (2007)
18. Weaver, B.C., Podhrasky, R.J., Nemat, A.: Metal detector for identifying target electrical characteristics, depth and size (1997)
19. Port, D.M., Burch, I.A., Deas, R.M.: Dual height metal detection for clutter rejection and target classification. In: Detection and Remediation Technologies for Mines and Minelike Targets VII, vol. 4742, pp. 746–755. SPIE, Orlando (2002)
20. Baum, C.E.: Detection and identification of visually obscured targets. Electromagnetics library. Taylor & Francis, Philadelphia (1999)
21. Watanabe, R., Sato, S., Tomita, S.: Method and apparatus for use in separation and recovery of non-magnetic metal pieces (August 15, 1978)
22. Tran, M.D.J., Abeynayake, C., Jain, L.C., Lim, C.P.: An Automated Decision System for Landmine Detection and Classification Using Metal Detector Signals. In: Finn, A., Jain, L.C. (eds.) Innovations in Defence Support Systems – 1. SCI, vol. 304, pp. 175–200. Springer, Heidelberg (2010)
23. Tran, M.D.J.: Automated Detection and Discrimination of Buried Explosive Threats Using Feature Extraction and Cost Based Probability Updating. PhD thesis, Knowledge Based Intelligent Engineering Systems (KES) Centre, School of Electrical and Information Engineering, Division of Information Technology, Engineering and the Environment, University of South Australia (August 2011)
24. Tran, M., Abeynayake, C., Jain, L.: A target discrimination methodology utilizing wavelet and morphological based feature extraction with metal detector array data. IEEE Transactions on Geoscience and Remote Sensing 50(1), 119–129 (2012)

How to Assess Human Visual Performance on an Operational Task

Katherine Hanton[1], Jadranka Sunde[2], Marcus Butavicius[3], Lakhmi C. Jain[1], and Nicholas Burns[3]

[1] School of Electrical and Information Engineering, University of South Australia, GPO Box 2471, Adelaide, SA, 5001, Australia
Katherine.Hanton@postgrads.unisa.edu.au,
Lakhmi.Jain@unisa.edu.au
[2] Defence Science and Technology Organisation, PO Box 1500, Edinburgh, SA, 5111, Australia
Jadranka.Sunde@dsto.defence.gov.au
[3] School of Psychology, University of Adelaide, SA
{Marcus.Butavicius,Nicholas.Burns}@adelaide.edu.au

Abstract. A methodology is presented for the assessment of human operator performance in a detection and identification task, using two sets of infrared images of natural outdoor scenes with everyday objects used as targets. It includes measures of effectiveness such as operator detection rate, identification rate, false alarm rate, response time, confidence levels and image quality ratings. This robust methodology could be used in the evaluation of any image improvement technique or to evaluate different imaging techniques or technologies.

Keywords: Infrared imaging, super-resolution, detection, identification, performance assessment, human operator.

1 Introduction

A number of technologies have potential for standoff detection [1, 2]. In this study, infrared (IR) imaging was chosen as it is capable of greater standoff distances and is portable. However, most currently available thermal imagers produce significantly under-sampled images, which have been found to negatively impact the detection performance of operators [3]. As such, the improvement of the resolution of IR images and the subsequent effects of the enhancement on the detection performance of operators is a worthwhile area of investigation.

In our application, we consider a scenario where an IR camera is attached to a vehicle moving through a natural outdoor environment, i.e., real world scenery. The human operator using the camera is required to detect objects of interest in natural scenes at a standoff distance. The standoff imaging distances chosen were 2.5, 5 and 10 m based on previously conducted studies (see [4]). Super-resolution (SR) image reconstruction was chosen as a method for improving the quality of the IR images.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 406–413, 2012.

The movement produced by an idling vehicle can be exploited to produce the images required for SR image reconstruction into high resolution IR images.

There has been a significant amount of research on SR with the majority of effort directed towards increasing the robustness and fidelity of the reconstruction process, with some work devoted to computational efficiency and real-time applicability [5]. However, an area that has been largely overlooked is the usefulness of SR in applications where the enhanced images will be subsequently judged by a human. Current image quality assessment methods are unsuitable for almost all applications involving a user detecting and identifying elements or objects in a scene [6]. In this paper, we propose and apply an empirical assessment approach, where the usefulness of SR processing is determined by the performance of human operators in specific detection and identification tasks.

1.1 Image Enhancement

In many applications, acquired images are often degraded by noise, blurring and aliasing effects caused by under-sampling imagers. This under-sampling has been found to have a negative impact on human performance when undertaking detection and identification tasks [3]. SR image reconstruction has been demonstrated to be very effective in removing these effects, and can also increase the spatial resolution of an image without changing the design of the optics or the detector. This is achieved by combining a sequence of low resolution (LR) images containing sub-pixel shifts to produce a higher resolution image [7-11]. In our work, the method of introducing controlled motion during image acquisition allowed the successful combination of 100 IR frames. While the most desirable method would be to exploit the natural movement produced by an idling vehicle, a diagonal dither technique was employed to introduce a controlled and repeatable movement during image acquisition [4, 12, 13].

1.2 Assessment of Image Enhancement

There are two conventional methods of assessing image enhancement techniques such as super-resolution [6, 14]. The first is by subjective assessment (often by the authors themselves) where the quality of the image is rated solely by visual examination of the images. However, the perceived quality of an enhanced image is not necessarily linked to the advantage it may provide in detecting or identifying objects, particularly when the image enhancement technique may increase the visual saliency of clutter or distractor objects. In fact, there is evidence that the effectiveness of displays in helping users perform these tasks may not necessarily be correlated with the perceived image quality [15]. The second approach to the assessment of image enhancement techniques is automated image quality metrics. While these may be more objective than image quality ratings provided by people, they have not been shown to correlate with the human visual system and the specific constraints associated with the user's expected task [14]. In this paper we adopt an empirical approach to the assessment of imaging techniques [16, 17]. This involves measuring the performance of a user as they attempt to use the images in a controlled task that is similar to the intended, final application, i.e., to detect and identify objects of interest in the images. In so doing, our performance measures better assess the performance of the SR technique with respect to its intended application than more conventional measures in the literature.

2 Methodology for Assessing Human Operator Detection Performance

To further establish under which circumstances SR image enhancement may have a positive effect on detection and identification performance, a study was designed to test if SR image enhancement may mitigate the negative impact of under-sampled images.

2.1 Description of Experiment

An experiment was designed to examine human operator detection and identification performance when using SR enhanced IR images. The scenarios considered were as follows:

- natural outdoor scenes with trees, leaves, man-made clutter;
- clear sunny day;
- three outdoor scenes with different levels of complexity (clutter) present;
- three imaging distances (2.5, 5 and 10 m); and
- eight targets of similar shape but different material and surface complexity.

Scenes imaged either had no target present, or contained one target object selected out of the eight possible targets. A total of 102 participants took part in the study, with 51 viewing only LR images, and the other 51 only viewing SR images.

Participants were asked detailed demographic questions regarding their age, if they had any experience with viewing/working with IR images/imaging, if they played computer games, their education level, their occupation, and if they had any vision problems. This information was collected in order to test for correlations between demographic variables and operator performance in the object detection task. They were also asked to complete post-test questions regarding the study and any difficulties they may have had to assist with the performance analysis.

Participants were given the same set of instructions and were given practice runs for each task where they were able to ask questions for clarification. In the experiment, participants were instructed to attempt to locate and identify the target present in the scene. The test images were presented to participants using purpose written software known as the ‘Object Detection and Identification Image Viewer’ (ODIIV), with a graphical user interface (GUI). This enabled control over how the images were presented and facilitated accurate logging of participant performance.

Prior to the commencement of image presentation, the required parameters for the ODIIV application need be set. These include the following:

- the 'File' menu for the selection of the folder containing the test images to be used;
- the 'Options' menu that allows control over the image presentation time (i.e., no time limit, or 60 second time limit);
- the type of image being viewed (i.e., bitmap, GIF, JPEG, Metafile or Icon); and
- the image size modes, i.e.;
  - 'normal' (original image size displayed from the upper left had corner);
  - 'stretch' (stretches image to fit GUI proportions);
  - 'zoom' (fits image to screen without aspect ratio distortions); or
  - 'centre' (as in 'normal' mode but displays from the centre of the GUI).

The image presentation time was set to a maximum time of 60 seconds for the duration of the study, with the test image scaled to fit the screen (i.e., 'zoom' mode) so that both LR and SR images were displayed at the same size.

Once the participants were instructed to select the 'Start' button, the first test image appeared. The participants were able to see an enlarged version of the target icons by hovering the mouse cursor over them, as shown in Fig. 1.

Fig. 1. ODIIV graphical user interface

If the participant decided that they had detected and identified one of the targets in the scene, they were instructed to drag the number associated with the target across to the test image and place it directly above the target position (shown in Fig. 2). A warning message would appear if the participant tried to move to the next image (by choosing 'Detection completed, move to next') without choosing one of the eight target objects, or 'No Targets'.

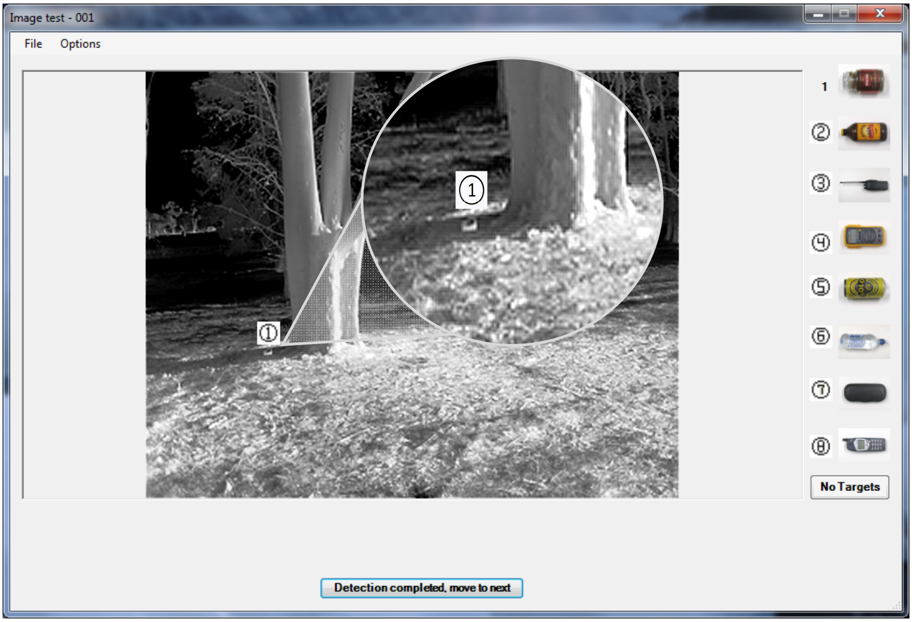


Fig. 2. GUI example with selected target

Once the participant selected 'Detection completed, move to next', the test image was replaced by a grey space, with three rating scales beneath it. The participant was then required to rate their confidence in their decision on a scale of 1 to 100 for detection and identification, as well as subjectively rate the image quality on a scale of 1 to 100 (Fig. 3). The image was removed from view to avoid influencing participants' confidence rating. Participants were not able to continue (i.e. 'Next image' button would not appear) until ratings had been provided for all three scales.

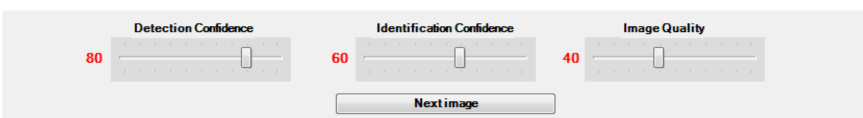


Fig. 3. Confidence rating GUI

The ODIIV saves a screen capture of the participants' detection selection and the time taken for their decision, as well as a record of their confidence ratings (for detection and identification separately) and their perceived image quality.

2.2 Performance Metrics

Performance was examined in terms of the combinations of standoff distance and target absent or present scenarios for correct, incorrect (false positives and false negatives) and timed-out responses:

- **detection rate** (correct responses) – a correct response indicates that the participant correctly declared that there was no target in the image (when the target was absent), or correctly located the position of the target (when the target was present);
- **identification rate** – participant correctly identified what the located target was;
- **false alarms rate** (incorrect response) – an 'incorrect' response indicates that the participant incorrectly: indicated that there was a target present when in fact it was absent; indicated that the target was absent when it was present; or indicated a clutter item / artefact as the position of a target. Incorrect responses were further split into false positive and false negative responses for target present images: a: **false positive** response indicates that the participant selected something other than the target in the image; and a **false negative** response indicates that the participant incorrectly selected the 'No Target' option;
- **response time** – how long each participant took to make their decision; 'timed-out' responses were also recorded, which indicate when a participant did not respond within the 60 second time limit;
- **confidence level** for each participant's detection and identification decision using a 100 point rating scale, where '1' means not confident at all and '100' means highly confident; and
- **image quality rating** using a 100 point rating scale, where '1' means very poor image quality and '100' means very good image quality.

2.3 Significance Testing

Human performance was analysed for each of the defined metrics. As there were two primary image groups under investigation (LR and SR), the Mann-Whitney *U* test was chosen for significance testing [18] and applied using the application XLSTAT [19]. The Mann-Whitney *U* test was chosen as it is the non-parametric alternative for comparing scores from two independent samples, where a statistical significance value of $p \leq 0.001$ is reported as being highly significant.

3 Results from Comparison of LR and SR IR Images

It is clear from this work that human operator detection performance is quite complex, and a number of factors may affect detection and identification performance. The results showed that, when examining overall performance in terms of imaging distance, SR was only of benefit to participants at 2.5 m. However, more detailed analyses of participants' responses showed that SR was able to improve performance only when the target was present. While there were no significant differences in performance when the target was absent for each of the three distances, the use of SR images did not lead to poorer performance either.

Performance was examined for target present images in terms of correct, incorrect, and timed-out responses at the three imaging distances. These results indicated that participants who viewed SR images at the 2.5 and 5 m distances had significantly better performance (for both correct and incorrect responses) than those viewing LR images, but not at the 10 m distance. While SR processing has the potential to improve operator performance, it may be that it only has limited utility. While the number of timed-out responses was relatively low, there was a significant difference in timed-out responses at the 5 m distance, where those viewing SR images had significantly more timed-out responses than those viewing LR images.

Additionally, for target present images, SR processing significantly increased the level of identification performance at the 2.5 and 5 m distances. However, at the 10 m distance, a significant difference in performance was found, where those viewing LR images had better identification performance than those viewing SR images. In summary, results from the human operator performance testing indicated that an apparent increase in image sharpness does translate into human performance gains, but that SR may not be universally advantageous, and that distance may affect the usefulness of SR.

The effect of participant IR experience and age were also explored. Negative correlations were found between participant performance and participant's experience and age. In both cases, performance decreased as experience and age increased.

4 Conclusions

While the methodology presented in this paper was developed for our specific application, it could be also be used to assess a variety of other types of images or imaging techniques. Although our empirical approach requires more effort than subjective assessments or automated image quality assessment techniques [6], the results produced using the methodology better reflect the usefulness of an evaluated technique in real-world applications.

Acknowledgements. The authors would like to thank Vido Gluscevic for writing the ODIIV software.

References

1. The National Research Council. Existing and Potential Standoff Explosives Detection Techniques. National Research Council ed. National Research Council. The National Academies Press, Washington, DC (2004)
2. Parmeter, J.E.: The Challenge of Standoff Explosives Detection, pp. 355–358. IEEE (2004)
3. Vollmerhausen, R., Driggers, R.G., O'Kane, B.L.: Influence of Sampling on Target Recognition and Identification. Optical Engineering 38(5), 763–772 (1999)
4. Hanton, K., Sunde, J., Butavicius, M., Burns, N.: Super-resolution of Infrared Images: Does it Improve Operator Object Detection Performance? Journal of Comp. and Inf. Technology (CIT) 18(2), 141–150 (2010)
5. Baker, S., Kanade, T.: Super-Resolution: Limits and Beyond. In: Chaudhuri, S. (ed.) Super-Resolution Imaging, pp. 243–276. Kluwer Academic Publishers, Boston (2001)

6. Wang, Z., Bovik, A.C.: Modern Image Quality Assessment, Synthesis Lectures on Image, Video and Multimedia Processing. Morgan and Claypool Publishers, USA (2006)
7. Tian, L., Suzuki, A., Koike, H.: Task-Oriented Evaluation of Super-Resolution Techniques. In: 20th International Conference on Pattern Recognition, Istanbul, pp. 493–498 (2010)
8. Li, J., Sheng, U.: Wavelet Domain Superresolution Reconstruction of Infrared Image Sequences. In: Proceedings of SPIE in Sensor Fusion: Architectures, Algorithms and Applications, vol. 4385, pp. 108–116 (2001)
9. Van Eekeren, A.W.M., Schutte, K., Oudegeest, O.R., Van Vliet, L.J.: Performance Evaluation of Super-Resolution Reconstruction Methods on Real-World Data. EURASIP Journal on Advances in Signal Processing, 1–11 (2007)
10. Choi, E., Choi, J., Kang, M.G.: Super-Resolution Approach to Overcome Physical Limitations of Imaging Sensors: An Overview. International Journal of Imaging Systems and Technology, Special Issue: High-Resolution Image Reconstruction 14(2), 36–46 (2004)
11. Park, S.C., Park, M.K., Kang, M.G.: Super-Resolution Image Reconstruction: A Technical Overview. IEEE Signal Processing Magazine 20(3), 21–36 (2003)
12. Hanton, K., Butavicius, M., Sunde, J., Lozo, P.: Operator Measures and Edge Sharpness Metrics to Assess Infrared Image Enhancement. In: Proc. of the 32nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 21–26. IEEE Press, Opatija (2009) (invited paper)
13. Hanton, K., Butavicius, M., Johnson, K., Sunde, J.: Improving Infrared Images for Standoff Object Detection. In: Proc. Information Tech. Interfaces (ITI 2009), pp. 641–646 (2009); also in Journal of Computing and Information Technology (CIT) Special Edition 18(2), 151–157 (2010)
14. Burningham, N., Pizlo, Z., Allebach, J.P.: Image Quality Metrics. In: Image Processing, Image Quality, Image Capture Systems Conference, pp. 598–616 (2003)
15. Frøkjær, E., Hertzum, M., Hornbæk, K.: Measuring Usability: Are Effectiveness, Efficiency, and Satisfaction Really Correlated? In: Proc. of Conference on Human Factors in Computing Systems, The Hague, The Netherlands, pp. 345–352. ACM, New York (2000)
16. Agarwal, S., Reddy, M., Hall, R., Woodard, T., Brown, J., Trang, A.: Evaluating Operator Performance in Aided Airborne Mine Detection. In: Proc. of SPIE Detection and Remediation Technologies for Mines and Minelike Targets X. SPIE, Orlando (2005)
17. Butavicius, M.A., Parsons, K.M., McCormac, A., Foster, R., Whittenbury, A., MacLeod, V.: Assessment of the ThruVision T4000 Passive Terahertz Camera: A Human Factors Case Study. In: Jain, L.C., Aidman, E.V., Abeynayake, C. (eds.) Innovations in Defence Support Systems -2. SCI, vol. 338, pp. 183–206. Springer, Heidelberg (2011)
18. Motulsky, H.: Intuitive Biostatistics. Oxford University Press, New York (2005)
19. Addinsoft, XLSTAT. Version 2011.4.02 (2011), `http://www.xlstat.com/`

Scenario Modeling and Verification for Business Processes

Bruno Blašković, Zoran Skočir, and Luka Humski

University of Zagreb,
Faculty of Electrical Engineering and Computing,
Unska 3, 10000 Zagreb, Croatia
{bruno.blaskovic,zoran.skocir,luka.humski}@fer.hr
http://www.fer.hr/

Abstract. In this paper we describe modeling for verification of business process with Spin model checker. Our primary goal is the development of Promela language description for e–invoice web service. Modeling for verification follows Church's synthesis problem: for input scenario "data–flow" model, output is Promela model. Sequence of model transformations translates the scenario into the Promela language code. At the end whole process is illustrated with e–invoice web service example.

Keywords: scenarios, model checking, model transformation, Spin model checker, model driven engineering, interworking formal methods, finite automaton.

1 Introduction

Business processes management [1] is one of the domains that needs more research in order to approach defined quality of service. Web based services that provides a support for the e-transactions are distributed application with the strong relation to communication processes in distributed systems [2]. In this paper, the goal is modeling for verification of e–transaction system known as e–InVoice. In the first place we describe the modeling process, while model checking analysis is out of the scope of this paper. Model checker "checks" or verify selected properties of the system. Selected properties are derived following properties known from concurrent software verification like *liveness* and *safeness*. Besides that, additional verification efforts are necessary for data–flow consistency between the entities in e–transactions. E–transactions involve rather simple operations but there are many (thousands) operations in parallel.

Model is the set of communicating entities that exchange messages. Sequences of messages from the model define the system behavior. Data–flows in the system and sequences of the messages from the model are two viewpoints of the same system. The process of modeling translates data flows to corresponding message sequences. For that purposes transition based formalism has been used. Finite state automaton, message sequence charts and UML diagrams are used to model the system behavior. Messages describe data structures interchanged between

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 414–423, 2012.

the processes. Starting from partial system descriptions known as scenario *ScD*, the Promela model is defined [3]. After that industrial strength tool Spin can analyze desired properties.

In ideal situation an agent $\mathcal{AG}$ will translate "data–flow" model directly to Promela model. In real situation this goal is achievable through series of model transformation. In this paper the term "model" $\mathcal{M}$ is used as a description of each transition diagram, for example $\mathcal{M}(FSA)$ or $\mathcal{M}_{FSA}$ for finite state automaton. Each model has internal representation that follows formal definition (for FSA or UML) or tool language (Promela) input language. Each message has internal information content that strictly follows XML data representation.

This paper is organized as follows: after introduction related work with theoretical background is presented in Sec. 2. Sec. 3 is the central Section of the paper. It presents our approach for Church's synthesis problem as domain specific model transformation. Basic definitions necessary for the next Sections together with interworking formal model framework structure are also included. Sec. 4 describes model $\mathcal{M}$ sequence chains in more details together with some illustrative diagrams. At the end in Sec. 5 conclusion with further research directions is presented.

2 Background and Related Work

Basic theoretical background is the definition of Church's synthesis problem introduced in Sec. 3. Scenario translation to Labeled transition system (LTS) is described in [4]. In [5] useful background for process mining is found. Besides the agent technology components [6] are also usable as models. Component approach is parallel to the ours, and we believe that agent and component approaches can exist together. That existence is possible because of [7] integration of formal models and model driven engineering [8].

Our model transformation is linear, while [9] introduces hierarchical automaton network. Previous work from scenario synthesis [10], [11] and protocol verification has implication [12] on definition of "triptych" environment (Sec. 3.2) and interworking of formal methods.

Besides industrial strength Spin model checker [3] Petri net models [13], [14],[15], [16] and [17] are widely in use for UML verification. Recent advancements are experimenting with *Yices* [18] SMT solver for model analysis.

3 Scope and Motivation: Agents, Components and Synthesis

This paper proposes synthesis approach for Promela model definition. Church's synthesis problem (Fig. 1) can be stated as follows:

"Given `<inputs>*` and `<outputs>` define the content Ψ of the box"

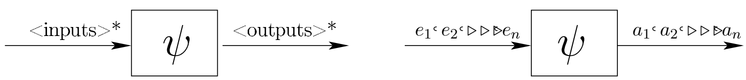


Fig. 1. Church synthesis and agents

In the agent $\mathcal{AG}$ context [19], [20] <inputs>* are sequences of events from environment $e_1, e_2, \ldots, e_i, \ldots, e_n$ and <outputs> are actions $a_1, a_2, \ldots, a_i, \ldots, a_n$. Every event and action has zero or more parameters $e_i(p_1, p_2, \ldots)$ and $a_i(p_1, p_2, \ldots)$, respectively. Content of the box is reactive agent because for each input event e_i action a_i as generated as a result. Reactive agent is modeled as transition system and formally defined as finite state automaton (FSA). Events e_i and actions a_i or are labels on transition in FSA. Internal FSA labels are represented as τ labels. Semantic of FSA transition is ${}^{e_i}/_{a_i}$ or for input event generate action as output event.

Ideal $\mathcal{AG}$ will define FSA in single step. Real $\mathcal{AG}$ will need several steps:

$$\mathcal{M}_{FSA_1}^{(\lambda_k 1)} \xrightarrow{\psi_{k+1}} \ldots \xrightarrow{\psi_{k+2}} \mathcal{M}_{FSA_2}^{(\lambda_k 2)} \xrightarrow{\psi_{k+3}} \ldots \qquad (1)$$

Each $\mathcal{M}$ from Eq. 1 is the formal representation of single *FSA* or the set of interacting *FSA*. (λ) in model superscript introduces the level of abstraction: FSA model can be "data–flow" model , intermediate FSA, or Promela program. Arrows from Eq. 1 present model translation. Model $\mathcal{M}$ translation consists of the sequential calls of algorithms ψ_k. Natural choices for $\mathcal{M}$ are UML [21] diagrams , SDL [22], with MSC [23] and model checking tool input languages (Promela, Petri nets, NuSMV). To realize proposed translation several methods from software engineering are used. After initial considerations, we define agent that must perform three actions (three="triptych"). Actions are "arrows" on Fig. 2 and events are presented as "data in circles". Practical evolution shows that generated Promela models are under expectations only when we introduce more actions. That leads to additional modifications that has resulted in the following model characteristics:

(1) every model $\mathcal{M}$ is formalized as FSA. Every viewpoints of the system has optimal formal or semi–formal model $\mathcal{M}$ representation consisting of "data–flow", UML diagrams and FSA models. Model transformation takes place between various model representations,
(2) synthesis problem always transforms starting representation into the final representation. As a feasible solution, we introduce more translation steps, rather then use of complex algorithms. Structure of the each model is described as declaration following formal definition. For example, translators exists for UML statechart translation with Spin tool or with Petri nets. As a side effect, the problem of formal methods interworking must be resolved, In some occasions, especially with non–formal specifications, that is not possible, and agent will perform the task of *proces mining.*

(3) models can have another viewpoint as component [6]. Model transformation is component composition, component transformation and component interworking, respectively. In this paper component models are not analyzed.

Agent technology is also useful for verification model definition of 1–state protocols but with many interacting ad–hoc entities. Each entity must work together with other in consistent way during whole life–time cycle. Input model is usually in semi–formal or narrative form, or built–in the source code with informal drawings. Promela program or Promela model is formal description of the problem.

When the system has no feasible formal specification, additional modifications, standard changes, maintenance and regression tests during system exploitation always degrade system's quality of service. In order to improve e–transaction

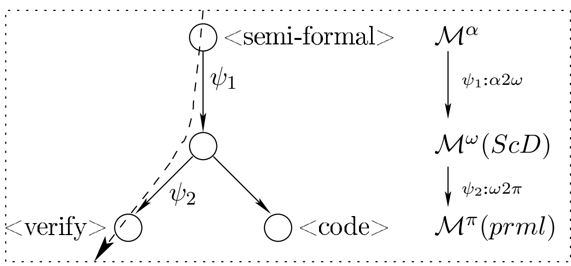


Fig. 2. Triptych "three–fold" general structure

modeling and verification process system development consists of the following steps (Fig. 2):

(1) $\mathcal{M}$ is agent that "understands" its environment.
(2) Agent is designed for verification of desired property: consistency data check and the support of test–cases as well as test–data generation,
(3) Intermediate representation for possible application skeleton generation.

That allows us to apply methods and tools for communication software design [12] for specification and implementation of business processes. Web based services and business processes have their special domain languages like BPEL [5].

3.1 Triptych "Three–Fold" Model Transformation Environment

From the reactive agent synthesis through model transformations point of view, for many distributed applications there is the same development environment (Fig. 2):

(1-fold) given scenario description ScD model (partial description of e–transactions) generate verification model:

$$\underset{<\text{ semi-formal }>}{\mathcal{M}^\alpha(ScD)} \xrightarrow{\psi_1} \mathcal{M}^\omega(FSA) \xrightarrow{\psi_2} \underset{<\text{ verify }>}{\mathcal{M}^\pi(prml)}$$

Superscripts in $\mathcal{M}$ describes level: α is scenario semi–formal level or abstract level, ω is intermediate or *FSA* level or operational level, π is Promela code level or implementation level. Previous protocol engineering and protocol synthesis [12] influenced such approach. Besides Promela models or programs, this environment is suitable for other tools inclusions like Petri nets [13], [16], [17], satisfiability solvers [18] and model checkers (NuSMV, SMV, UPPAAL, ...),

(2-fold) given scenario description ScD model $\mathcal{M}(ScD)$ generate code skeletons. This is automated program synthesis or automated programing. This is not under considerations in this paper,

(3-fold) given formal model description model $\mathcal{M}^{\alpha}(FSA)$ verify system properties. This model transformation describes automated test generation (ATG), model checking and (eventually) theorem proving.

Classical approach for non–reactive system development usually starts from node `<semi-formal>` on Fig. 2 and directly finishes in the node `<code>`.

3.2 Interworking of Formal Methods for Model Transformation

Besides model transformation, realization of algorithms requires sound and complete logical definition for each model . Upper limit requires the same number of formalism for the each model $\mathcal{M}$. Another possibility is multiagent system MAS where each model transformation ψ_k has corresponding agent $\mathcal{AG}_k(\psi)$.

As a solution to this dilemma, hybrid approach is realized yielding optimal solution. All models on all level are described as transition system. Formally, each model $\mathcal{M}$ is finite state automaton (FSA). FSA is "recognized" in UML or MSC diagrams as well as in target implementation code (like `C`, `Java` or `C++`). Promela program C(FSA) is the set of communicating FSA.

Only the first model translation from "data–flow" model ($\mathcal{M}(DtFl) \xrightarrow{\psi_1}$) is not possible with hybrid approach. Table 1 presents the relations between agent, implementation, business process methodology and formal methods:

Table 1. interworking of formal methods for triptych environment

agent	implementation	bpm	math:FM
$\mathcal{AG}$	`proctype()` in Promela	process UML.SqD	FSA,MSC
MAS	set of proctype (prml prgm.)	system ScD (UMLAct)	C(FSA)

- each column describe the viewpoint of the model $\mathcal{M}$. For example, model M is Promela model $\mathcal{M}(prml)$ or $\mathcal{M}(FSA)$ is finite automaton FSA ...,
- the first row describes model M on unit level. For example `proctype()` language construct in Promela language is basic unit of modeling in implementation, *UML* sequence diagram is basic unit of modeling in business process,
- the second row describes model M on system level. For example, during implementation system is modeled as the set of `proctype()` Promela constructs or as a communicating *C(FSA)* in formal model FM.

3.3 Preliminary Definitions

Scenario [4], is partial description of system behavior.

Business proces model is workflow model [1] for support, definition, execution, registration and control of processes. Business process is modeled with BPEL language [5] or other formal models like Petri nets.

Agent is [19] high-level description of the goal, with built–in mechanisms to figure out what to do, knowing that agent will act in accordance with some built-in theory.

FSA is 5–tuple (s_0, S, T, L, F) or `FSA` is the model $\mathcal{M}$ such that:

s_0 is initial state,
S is the set of places,
T is the set of transitions, $S \cap T = \emptyset$,
L is the set of labels,
F is the set of final states

4 Models and Transformations

Starting from semi–formal "data–flow" model $\mathcal{M}_{DtFl}$ from Fig. 3 sequence diagram $\mathcal{M}_{msc}$ is derived. "Data–flow" like chart on Fig. 3 identifies three documents denoted as gray circles, laying on "time–line" between black–box from Fig. 1 inputs and outputs. These documents are basically the same as in usual transaction carried outside e-InVoice class of services:

(1) provider "provides" e-InVoice document to the user,
(2) user "pays the bill" or user prepare document that are requests for provider to execute transaction,
(3) user is informed about transaction results.

All documents are XML files, with proper structure and data. Documents preparation takes several steps with the numerous amount of internal transactions on user's and provider's sides respectively. Transactions are carried out with the support of databases, applications and network infrastructure. They are represented as arrows. White circles on Fig. 3 represent internal data between internal transactions on both sides.

Transformation $\psi_{10} : \alpha 2\alpha$ is also presented on Fig. 6. Currently transformation is performed manually, as designer defines sequence diagrams directly from scenario (*ScD*) description. Intelligent process mining agent will automatize the work. However this is separate research direction.

Note that there are two sequence diagrams: MSC $\mathcal{M}(msc)$ and UML sequence diagram $\mathcal{M}(SqD)$:

- MSC (Message Sequence Diagram) [23] have better expressiveness in message exchange and concurrency issues: labels in MSC are messages,
- UML [21] sequence diagram (*UML.SqD*) are closer to the source code: labels in `UML.SqD` are objects.

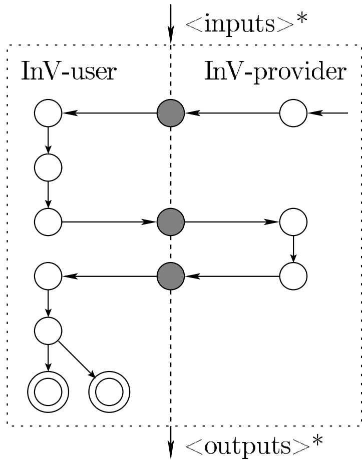


Fig. 3. Data flow semi–formal representation of scenario

Message exchange has been separated in order to avoid complex translation algorithms ψ_{11} and ψ_{12}. In such way model transformation between $\mathcal{M}_{FSA}$, $\mathcal{M}_{C(FSA)}$ and $\mathcal{M}_{prml}$ are performed over textual files. Examples of textual files representation can be seen on in Fig. 6 as gray (shadowed) area.

The modeling process consists of six steps, there are five ψ labels on Fig. 6. In such way model transformations and tool support yields optimal interworking of formal methods.

Slightly simplified example is presented on Fig. 4. The first iteration has been performed without agent technology support:

(1) the model transformation starts the top–level (called α level) of a scenario $\mathcal{M}(ScD)^\alpha$ with XML documents as labels ($l_i \equiv m_j \equiv e_k$). $\mathcal{M}(ScD)^\alpha$ is modeled as sequence diagram ($\mathcal{M}(SqD)^\alpha$),
(2) the translation from sequence diagram to FSA is straightforward: FSA states are inserted between messages. Messages from $\mathcal{M}(SqD)^\alpha$ and labels from $\mathcal{M}(FSA)^\alpha$ are equivalent. Event e_i is from the input sequence of events (Fig. 1),
(3) translation $\psi_{20} : \alpha 2\omega$ to communicating FSA ($\mathcal{M}(C(FSA))^\omega$) and to Promela model $\psi_{30} : \omega 2\pi$.

As a result, error messages for irregular process termination (`unreached in proctype pP1`) has been reported from Spin model checker. In order to correct design error two approaches are possible: adjust every FSA to behave like reactive system or introduce new end–states. The first solution was not feasible: reactive system will always reach desired end–state but in some occasion will cancel regular procedure. For that purpose translating algorithms are modified as agents: from top–level scenario ($\mathcal{M}(ScD)^\alpha$) end–state conditions are collected. After that, the calls to translation algorithms are expanded with the rules for the addition of regular and irregular end–states.

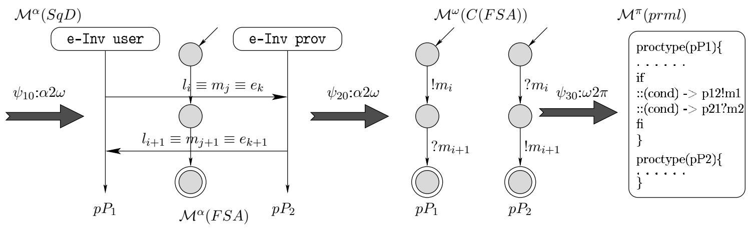


Fig. 4. Model Transformations Example

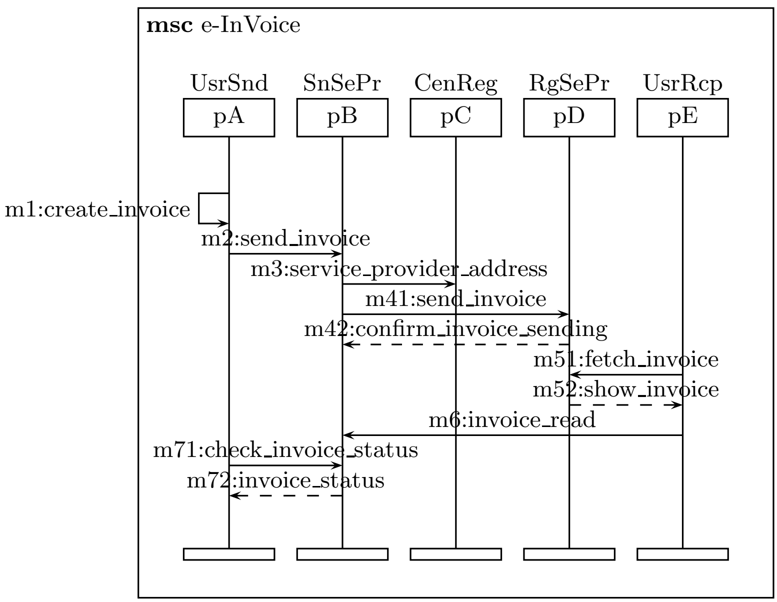


Fig. 5. ScD Scenario description as message sequence chart $\mathcal{M}_{msc}$

5 Conclusion and Further Research Directions

We have presented solution for e–InVoice Promela model definition. Our solution generate Promela program from scenario description by the means of sequence of model transformations and with the help of interworking formal models. We believe that model driven engineering must have clear and selected goal on selected domain. In our case the goal is e–InVoice model checking yielding the discovery of bugs and inconsistencies in the earliest possible phase. Generalization to "all" problems and domains yields no feasible result. To our knowledge similar e–transactions or web based services are solvable within proposed approach. Our results can be summarized as follows:

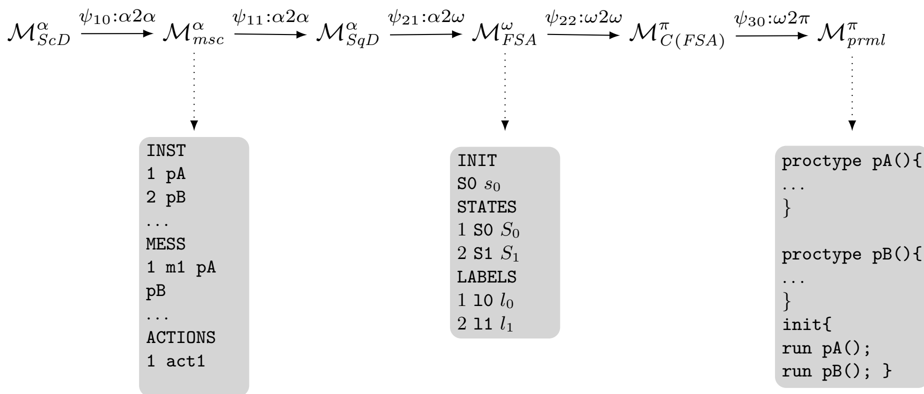


Fig. 6. The Chain of Model Transformations

- formalizing of the design process as Church synthesis problem,
- inclusion of tools for performance modeling or network traffic simulation,
- system is open for modifications, for example, for changes in international accounting standards, revision of banking e–documents as well as network infrastructure,
- regression testing: this part is always bottleneck of development. Promela model provides enough information to generate test–data.

Further work as well as research directions should concentrate on the following topics:

- the definition of process mining agent. This is promising research direction together with automated formal methods and software synthesis,
- experiments in related software engineering approaches like component engineering, artificial intelligence with model checking, and search based software engineering,
- test–cases and test–data automatized generation directly from scenarios,
- skeleton code definition with `assert()` and `assume()` statements for software model checking,
- automated software engineering: UML as implementation language or executable specification.

References

1. van der Aalst, W.: The application of petri nets for workflow management. The Journal of Circuits, Systems and Computers 8(1), 21–66 (1998)
2. Milner, R.: A Calculus of Communicating Systems, vol. 92. Springer, Berlin (1980)
3. Holzmann, G.: Spin model checker, the: primer and reference manual, 1st edn. Addison-Wesley Professional (2003)

4. Uchitel, S., Brunet, G., Chechik, M.: Synthesis of partial behavior models from properties and scenarios. IEEE Transactions on Software Engineering 35(3), 384–406 (2009)
5. OASIS Standard, Web Services Business Process Execution Language Version 2.0 (2007), http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html
6. Crnkovic, I., Sentilles, S., Vulgarakis, A., Chaudron, M.R.V.: A classification framework for software component models. IEEE Transactions on Software Engineering 37(5), 593–615 (2011)
7. Grosse-Rhode, M.: Semantic integration of heterogeneous formal specifications via transformation systems (2001)
8. Caskurlu, B.: Model Driven Engineering. In: Butler, M., Petre, L., Sere, K. (eds.) IFM 2002. LNCS, vol. 2335, pp. 286–298. Springer, Heidelberg (2002)
9. Whittle, J., Jayaraman, P.K.: Synthesizing hierarchical state machines from expressive scenario descriptions. ACM Trans. Softw. Eng. Methodol. 19(3) (2010)
10. Blaskovic, B., Randic, M.: From declarative model to solution: Scheduling scenario synthesis. In: 9th International Conference on Telecommunications, ConTel 2007, Zagreb, Hrvatska, June 13-15, pp. 139–142 (2007), doi:10.1109/CONTEL.2007.381862.
11. Blaskovic, B., Randic, M.: Model Based Scheduling Scenario Generation. In: Katalinic, B. (ed.) DAAAM International Scientific Book, pp. 031–044. DAAAM International, Vienna (2006) ISBN 3-901509-47-X, ISSN 1726-968704, doi: 0.2507/daaam.scibook.2006.04
12. Blašković, B., Lovrek, I.: Verification of Signalling Protocols for Telecommunication Services in Intelligent Network Environment. In: 1st International Workshop on Applied Formal Methods in System Design, Maribor, pp. 156–165 (June 1996)
13. Murata, T.: Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE 77, 541–580 (1989) (an invited survey paper)
14. Bhaduri, P., Ramesh, S.: Model Checking of Statechart Models: Survey and Research Directions, CoRR, cs.SE/0407038 (2004)
15. Korenblat, K., Grumberg, O., Katz, S.: Translations between Textual Transition Systems and Petri Nets. In: Butler, M., Petre, L., Sere, K. (eds.) IFM 2002. LNCS, vol. 2335, pp. 339–359. Springer, Heidelberg (2002)
16. Varpaaniemi, K.: Efficient Detection of Deadlocks in Petri Nets. Series A: Research Reports 26. Helsinki University of Technology, ESPOO, Finland (October 1993)
17. Gronberg, P., Tiusanen, M., Varpaaniemi, K.: PROD - A Pr/T-Net Reachability Analysis Tool. Series B: Technical Reports 11. Helsinki University of Technology, ESPOO, Finland (June 1993)
18. Dutertre, B., Moura, L.D.: The YICES SMT solver. Tech. Rep., SRI (2006)
19. Wooldridge, M.J.: Introduction to multiagent systems. Wiley (2002)
20. Shoham, Y., Leyton-Brown, K.: Multiagent Systems - Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press (2009)
21. Omg unified modeling language^{TM} (omg uml), OMG Document Number: formal/2009-02-04, Infrastructure Version 2.2 (2009)
22. ITU, Z.100 Specification and Description Language (SDL). ITU-T Telecommunication Standardisation Sector, Geneva (August 2002)
23. ITU, Z.120 Message Sequence Charts (MSC). ITU-T Telecommunication Standardisation Sector, Geneva (April 2004)

Agent-Based Analysis and Detection of Functional Faults of Vehicle Industry Processes: A Process Mining Approach

Ágnes Werner-Stark and Tibor Dulai

Department of Electrical Engineering and Information Systems,
Faculty of Information Technology, University of Pannonia, Veszprém, Hungary
{werner.agnes,dulai.tibor}@virt.uni-pannon.hu

Abstract. The technologic processes are executed in the recent technological systems by process directing devices. In these systems the instruments fix the measured values and the corresponding dates (timestamps) on the one hand, on the other hand it is also recorded who and what performed what kind of intervention during operation. The goal of our research is to work out special agents for resulting data from monitoring of functioning of discrete event systems with special regard to diagnosis of faulty mode of operation. In our research we examined vehicle industry as to what useful information can be filtered out for the sake of functional and cost efficiency from the logs resulting from the processes. One of the areas is one of the testing processes in vehicle industry which follows planning, and the other one is the examination of specific troubleshooting process in case of buses.

Keywords: agent, process, modeling, process mining, vehicle industry.

1 Introduction

The technologic processes are executed in the recent technological systems by process directing devices. In these systems the instruments fix the measured values and the corresponding dates (timestamps) on the one hand, on the other hand it is also recorded who and what performed what kind of intervention during operation. If we put the information in the correct order then we can get about the "history" namely course of the technologic process. In case of intermittent operation technology systems: in the course of producing of a piece of work or executing of product fixed data is called a trace. The traces we get during an unit of time, e.g. during one shift or one day, are called logs.

In case of intermittent operation technology systems - while e.g. we closely follow the production of one piece finished product - there is a close connection between during production fixed data and the quality of product. If this product is produced in large rate or in last meaningful time series production then we can prove tendencies too by examination of series of data, and so we can prevent possible deleterious effects.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 424–433, 2012.

The goal of our research is to work out agents for resulting data from monitoring of functioning of discrete event systems with special regard to diagnosis of faulty mode of operation. The course of diagnostic procedures can be manifold, but the methods have common property, we have to make the model of the study system. These models are reckoned as form usually different automaton or Petri net type. In both cases the model emphasizes the events taking place in the system and the preconditions respectively consequences of those. It is important to note that these models contain static and dynamic information for the system too. The structure of the model fixes static way in the system principle possible processes and the root causes, respectively consequences. At the same time we can follow closely current state of the system by the simulation of model, namely the given initial state started examination we can study what kind of intermediate state get through the system to the valid steady state on the shutdown. The idea of applying multi-agent systems [5] for diagnosis in engineering applications is becoming more and more popular, especially in manufacturing applications [15], [10]. An agent based diagnostic system has been proposed in [3] that uses HAZOP-FMEA information and bidirectional reasoning to derive a diagnostic conclusion [8].

In our research we examined vehicle industry as to what useful information can be filtered out for the sake of functional and cost efficiency from the logs resulting from the processes by software-agent. The area is the examination of specific troubleshooting process in case of buses.

Section 2 shows the background of our approach and introduces some concepts of diagnostic methods based on comparison of models and logs, then section 3 shows the most important properties of process mining. Section 4 shows the modeling, the agent based development and analysis results by means of a case study, while in section 5 we conclude our work.

2 Diagnostic Methods Based on Comparison of Models and Logs

In our section we deal with diagnostic methods, these can compare models and logs in different ways, because the developed agent - that we use in case study - compares the event log and the model of vehicle industry process.

2.1 Comparison of Models

To this, knowing the course of the process under normal conditions, we build up that model. If we have information for the possible faulty operation then we will make model too in compliance with those. Based on the data deriving from the monitoring of the process we execute the model that describes the actual operating, and then we define operating character by the comparison of models. The research work is duplicated in this case. One important step is how we can produce the corresponding operation model from data resulting from real processes. The other task is the comparison of models which can be done with comparative structure of graph algorithms. [6]

2.2 Comparison of Logs

Another diagnostic option is the comparison of logs. In this case we make a comparison between the technological logs got from production and the logs produced by means of the simulation of models. There is a possibility here too that we make comparison only between the log resulting from the correct operation descriptive reference model and the data of realized process or we take into consideration logs got from faulty models. At the foregoing method and in this case also if we make the comparison only on the basis of the normal operation descriptive model and the real operation descriptive data, then as a result the correct operation of the process or deviation from the fact can be diagnosed. If we take the faulty operation descriptive models into consideration then we can get information on the quality of fault as well. However, in some cases this only occurs if the fault has reached a certain level in the correct measurement data. [13]

2.3 Conformance Checking

Conformance testing means a kind of combination of the previous two diagnostic methods. In this case we make a comparison of the reference model and the set of logs resulting from the operation of the process. On the one hand we examine how the reference model fits into logs i.e. on the basis of that what proportion of logs could be produced, on the other hand how the structure of the reference model is suitable for the logs. [12], [14]

3 Process Mining

The economic and industrial processes, the software, the information systems have become quite complex, complicated and impenetrable. For these reasons, today's modern IT systems (e.g. WFM, ERP, CRM, SCM, and B2B systems) fix the events in log files. These stored logs contain data about executed activities; time stamps have reference to execution of tasks; individuals or systems, who and what executed the tasks etc. Due to these we have an opportunity to examine deepest points of the processes and to detect the possible arising problems, analyze and make the process itself cost-effective. The goal of the technology: make a map, supervise, control, and perhaps correct the real processes by extracting new information from the event logs.

Agrawal and associates [1] were the first who applied process mining in relation with business processes. Van der Aalst and associates dealt with researching of business processes in their works too (see e.g. [2], [12]). Earlier we already dealt with the possibility of process mining, fault diagnostic [13], and model structure identification [14] in some of our works.

3.1 Properties of Process Mining

In studies, we can get answers to many questions as e.g.: How are the events suitable for the made process model? Where are the problems? How common is

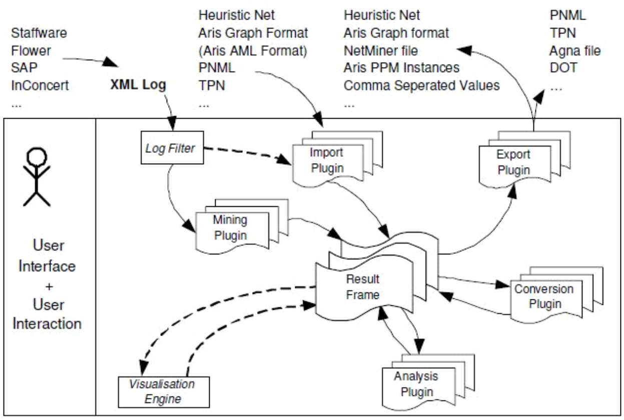


Fig. 1. Overview of the ProM framework (Source: [4])

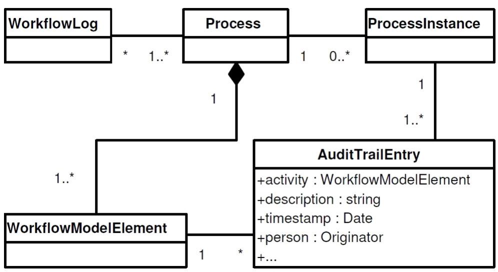


Fig. 2. Process log MXML format (Source: [4])

(not) there correspondence? What are the average minimum/maximum execution times of events? What is the average processing time of the certain tasks? What are the rules in the business process model? How many people are concerned at an event? et cetera.

3.2 Application of ProM Framework for Process Mining

The ProM framework [9] is an open source process mining tool (Fig. 1). The device can handle the most important process models as an input, e.g. the Petri nets. Further many plug-ins can be available to the application respectively there is an opportunity to develop and extend these plug-ins. An important property of it is that as input beside process models it is able to handle the logs in a specific MXML format (Fig. 2).

4 The Modeling and Analysis of the Process Related to the Repairing of Motor Vehicles

4.1 Description and Modeling of Process

The vehicle repair is a complex activity which can restore the good working of out of service or worn out parts, mechanical unit or main part belonging to the vehicle. The repair industry can be characterized mostly as a process chain. In this context, the input of the system is faulty; perhaps an out of service vehicle, the output is the vehicle which is developed as the result of processes within the system. At this point the importance of process mining appears too, by which problems within the system can be cleared up (the wrong order of some process), which are responsible for the high costs and the prolonging repairing time.

From the operation groups belonging to the bus repair and maintenance activities (weekly status examination; Number I. technical inspection; Number II. technical inspection; examination of participant buses in international traffic; technical test of vehicle preparation; extra inspections occasional failures to improve; repair under guarantee) we emphasize in here Number I. technical inspection. This process is important from the point of view of traffic and security of buses. This includes the examination of the structural details and main parts. The inspection periods are determined to the survey of the oil change limits. The limit should be strictly observed.

The Number I. technical inspection can be fissionable which happen linearly in a row. The first is the lower examination of the vehicle (LOWER EXAMINATION), that includes the controlling of the chassis, the motor, the undercarriage, the absorption of shocks, the shock absorber, the braking system, the control assembly, the gears, and the exhauster. The second group is the examination of tires, the body of wheels (TYRES CONTROL).

The third group contains the state survey of the upper examination of vehicles (UPPER EXAMINATION), including the accumulator, the electric wires, the fuel system, the driving compartment, the passenger compartment, the trunk, the heater, the lighting and the indicator. Completions of these generally take up $20 - 30$ minutes. For the bus to get into the repair industrial unit, at first it is needed to register at the working register (REGISTER BUS), then the bus is driven at the industrial unit (BUS TO SERVICE). After the revision (REVISION CHECKING) there are two possible routes. The first is that the inspector finds everything correct and the vehicle can be allowed to the traffic (REVISION COMPLETE and BUS TO TRAFFIC). Another way is that the inspector detects a fault. At this time there is nothing else to do; the bus has to be transferred to the respective service (GO TO SERVICE), (GENERAL SERVICE, TYRES SERVICE, ELECTRICITY SERVICE). After the service the re-revision process comes to pass (REVISION CHECKING). If every fault is repaired the vehicle will be ready to put in the traffic (REVISION COMPLETE and BUS TO TRAFFIC). If, however, after the second revision faults emerge, before putting in the traffic the vehicle is sent to the respective industrial unit (GENERAL SERVICE, TYRES SERVICE, ELECTRICITY SERVICE).

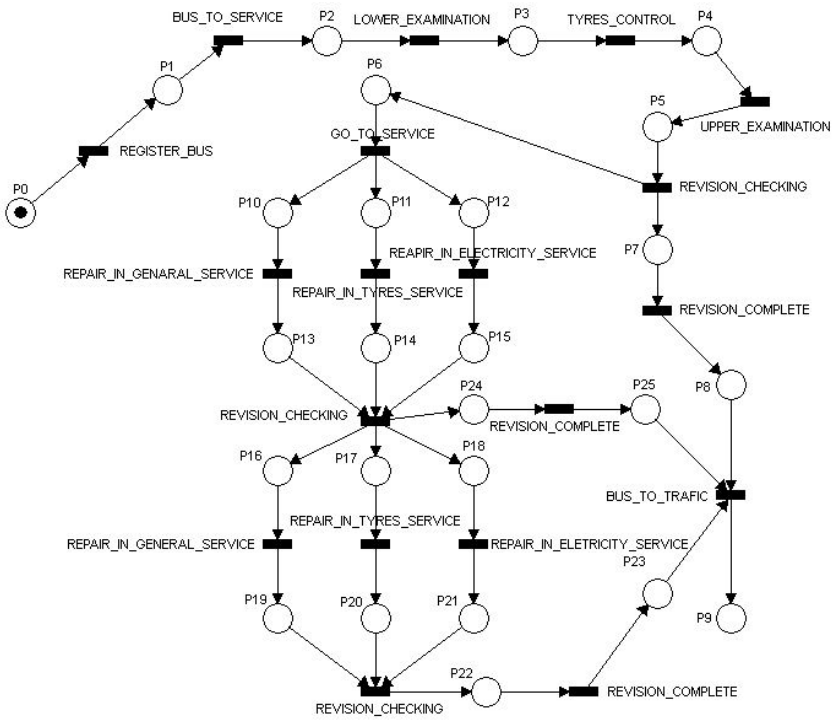


Fig. 3. The Petri net model is implemented in HPSim software

This process is modeled by us and the Petri net model can be seen in Fig. 3. A Petri net [7] emphasizes the possible states and the events occurring in the investigated system. A Petri net consists of places and transitions, and describes the relations between them. Transitions can be seen as the steps of operating procedures while places imply the preconditions and consequences of these steps in this controlled discrete event system.

4.2 Examining of the Model and Log with the Help of Existing Framework

We can use our MXML extension event log (Fig. 4), implementing our model in the ProM process mining system. We can control e.g. with the application of α-algorithm that how much the model in Figure 3 is suitable for the produced model from log. In the first place for that comparison is simple because the output of the α-algorithm [9] is Petri net. If there is agreement then we can conclude that our event log is correct and the log contains relevant data to the projected process.

For example, the *multi-phase plug-in*, the *Fuzzy Miner plug-in*, the *Originator by task Matrix* and the *Log Summary*, which can be found in ProM mentioned

```xml
<ProcessInstance id="11" description="Simulated process instance">
<AuditTrailEntry>
   <WorkflowModelElement>register_bus</WorkflowModelElement>
   <EventType>complete</EventType>
   <TimeStamp>2010-11-08T09:13:00.000+01:00</TimeStamp>
   <Originator>revisor1</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
   <Data><Attribute name = "busType">VolvoB12B </Attribute></Data>
   <WorkflowModelElement>bus_to_service</WorkflowModelElement>
   <EventType>start</EventType>
   <TimeStamp>2010-11-08T09:13:30.000+01:00</TimeStamp>
   <Originator>driver11</Originator>
</AuditTrailEntry>
<AuditTrailEntry>
   <WorkflowModelElement>bus_to_service</WorkflowModelElement>
   <EventType>complete</EventType>
   <TimeStamp>2010-11-08T09:16:00.000+01:00</TimeStamp>
   <Originator>driver11</Originator>
</AuditTrailEntry>
```

Fig. 4. Detail of the event log in MXML format

in sub-section 3.2, provided useful information about the process in the same way than in the earlier example.

4.3 Agent Development for the Analysis of the Auto Repair Process

Our goal was to create a decision agent that can be used for decision making. In artificial intelligence, an intelligent agent is an autonomous entity which observes through sensors and acts upon an environment using actuators (i.e. it is an agent) and directs its activity towards achieving goals ([11]). An agent is often described as a computer program. The basic attributes of a software agent are that

– agent may reside in wait status on a host, perceiving context,
– agent may get to run status on a host upon starting conditions,
– agent does not require interaction of a user,
– agent is able to develop proposals for decision-making with using of the described rules.

We developed a software agent based on its properties.

We studied the .mxml extension event log underlying process mining here too that was the sensible input to the agent. We divided the event log into two parts which were the most important from the point of view of our examination: One is the Processes part which stores the processes of the system. In this case we can get answer for the number of the events loaded in the logs. By selecting one we can show all the *AuditTrailEntry* belonging to the given process. By selecting one from these we can represent all the properties belonging to the given *AuditTrailEntry*. We can see the name of the process element the starting and the ending times as well as the name of the originator. (Fig. 5)

The other is the Originator part. After the selection of a concrete Originator it can be represented in which process or processes the given person was included. It can be decided how long he did the given operation using this information it can be determined how long the Originator worked altogether.

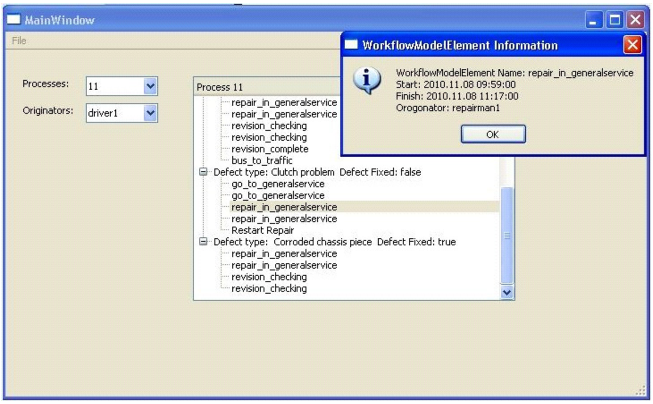


Fig. 5. The error level and correction of the "11" workflow (detail)

Let's examine a case in which at the bus faults emerged after doing the usual revision operations. At first the *revisor* detected a serious *Clutch problem* and he sent the bus to a general service (*repair in generalservice*). By selecting this event we can get to know that the bus passed 78 minute in this industrial unit. After correcting the fault, followed a re-inspection (*revision checking*), which diagnosed that more faults have occurred (corroded chassis parts), even in such form that the repairing is needed to be done in the general industrial unit (*restart repair*). Therefore the vehicle was taken again to the general service. After the completion of tasks, if the *revisor* haven't found faults, the vehicle can be placed in traffic.

Comparing the different process instances it can be concluded that a single revision process takes up about 30 minutes. If problems happen in the course of inspection the vehicle spends from 1 to 4 hours in the industrial unit depending on the seriousness and the number of faults.

But we have possibility to examine the Number I. technical inspection process not only from the point of view of individual cases but also from the point of view of the performing persons too. By selecting the originator we would like to examine, we can find:

– A given originator is included in process cases provided with what ID
– What events are performed in some process cases
– When he began, and completed his activity in a given event
– Total working time

The *repairman1* is chosen from the originators, it can be seen in Fig. 6., that he completed activity e.g. in *Process5,-7,-8,-11* process cases, that is to say, then the technician in the general industrial unit was needed. The mechanic worked total time 07 : 29 : 00 on the various vehicles. Similarly, we can control every originator, and then we can establish which person completed the most or just the longest activity series.

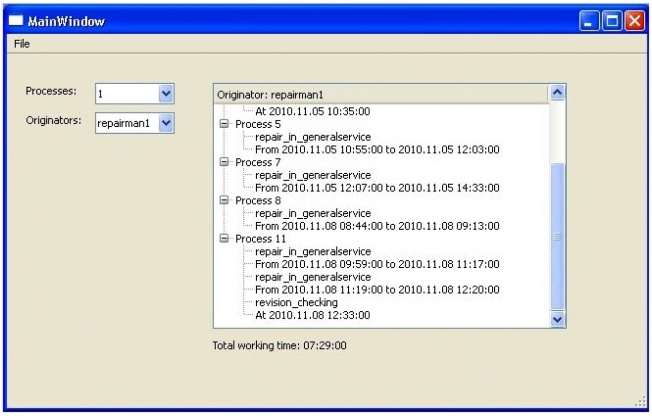


Fig. 6. Activities of repairman1 (detail)

The agent can give some decision proposal using this information to the industrial unit leader faster turnaround time to achieve.

5 Conclusion

In this paper we proposed a process mining approach for a problem of vehicle industry. When we examined the applicability of process mining in the field of vehicle industry, then we came to the conclusion that there are some implemented algorithms too which can be used e.g. in ProM system to the vehicle industry problem, but we couldn't extract all the important information what we needed in this special area.

That's why we modeled the process of the repairing of motor vehicles and analyzed this model and data that we could extract from process (these were log) in ProM system, but we needed further information. That's why we developed a special agent for process mining purposes which is capable to extract useful information from the process data and model. This agent was a decision agent that can be used for decision making in organize and control of the repairing process. In this paper we focused on some elements of the program that presents some special functions. Different conclusions can be concluded by the studies, e.g. comparing the different process instances it can be concluded that a single revision process takes up some minutes. This means that we can improve the execution time of the process.

It can be concluded that the agent can give some decision proposal using this information to the industrial unit leader achieving faster turnaround time.

We have shown that by the aid of the mined information the execution of the process can be made more efficient, and reduces the necessary costs for the execution of the process.

In the future our work can be extended by modifying the functions of software agent to further reduce costs.

Acknowledgment. We acknowledge the financial support of this work by the Hungarian State and the European Union under the TAMOP-4.2.1/B-09/1/KONV-2010-0003 project.

References

1. Agrawal, R., Gunopulos, D., Leymann, F.: Mining Process Models from Workflow Logs. In: Schek, H.-J., Saltor, F., Ramos, I., Alonso, G. (eds.) EDBT 1998. LNCS, vol. 1377, pp. 469–483. Springer, Heidelberg (1998)
2. van der Aalst, W.M.P., et al.: Business Process Mining: An Industrial Application. Information Systems 32(5), 713–732 (2007)
3. Cameron, I.T., Seligmann, B., Hangos, K.M., Lakner, R., Németh, E.: The P^3 formalism: A basics for improved diagnosis in complex systems. In: Proc. of the Chemeca Conference, Melbourne, Australia, p. on CD (2007)
4. van Dongen, B.F., de Medeiros, A.K.A., Verbeek, H.M.W(E.), Weijters, A.J.M.M.T., van der Aalst, W.M.P.: The ProM Framework: A New Era in Process Mining Tool Support. In: Ciardo, G., Darondeau, P. (eds.) ICATPN 2005. LNCS, vol. 3536, pp. 444–454. Springer, Heidelberg (2005)
5. Ferber, J.: Multi-Agent System: An Introduction to Distributed Artificial Intelligence. Addison Wesley Longman, Harlow (1999) ISBN 0-201-36048-9
6. Gerzson, M., Leitold, A., Hangos, K.M.: Model based Process Diagnosis using Graph Methods. In: Factory Automation 2011 Conference, Györ, Hungary, pp. 62–70 (2011)
7. Hangos, K.M., Lakner, R., Gerzson, M.: Intelligent Control Systems: An Introduction with Examples. Kluwer Academic Publisher, New York (2001)
8. Lakner, R., Németh, E., Hangos, K.M., Cameron, I.T.: Multiagent Realization of Prediction-Based Diagnosis and Loss Prevention. In: Ali, M., Dapoigny, R. (eds.) IEA/AIE 2006. LNCS (LNAI), vol. 4031, pp. 70–80. Springer, Heidelberg (2006)
9. de Medeiros, A.K.A., Weijters, A.J.M.M.: ProM Framework Tutorial. Technische Universiteit Eindhoven, The Netherlands (2009)
10. Micalizio, R., Torasso, P., Torta, G.: On-line monitoring and diagnosis of multi-agent systems: a model based approach. In: Proc. of the 16th European Conference on Artifical Intelligence (ECAI), Valencia, Spain, pp. 848–853 (2004)
11. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice Hall, Upper Saddle River (2003) ISBN 0-13-790395-2
12. Rozinat, A., van der Aalst, W.M.P.: Conformance checking of processes based on monitoring real behaviour. Information Systems 33, 64–95 (2008)
13. Werner-Stark, A., Gerzson, M., Hangos, K.M.: Model-Based Fault Detection and Isolation using Process Mining. World Academy of Science, Engineering and Technology 7(73), 851–856 (2011)
14. Werner-Stark, A., Gerzson, M., Hangos, K.M.: Discrete event model structure identification using process mining. In: Proceedings of the IASTED International Conference Modelling, Identification, and Control (MIC 2011), Innsbruck, Austria, pp. 228–233 (2011) ISBN 978-0-88986-863-2
15. Worn, H., Langle, T., Albert, M., Kazi, A., et al.: DIAMOND: Distributed multi-agent architecture for monitoring and diagnosis. Production Planning and Control 15, 189–200 (2004)

Diverse Sentiment Comparison of News Websites over Time

Jianwei Zhang[1], Yukiko Kawai[1], Tadahiko Kumamoto[2], Shinsuke Nakajima[1], and Yuhki Shiraishi[1]

[1] Kyoto Sangyo University
[2] Chiba Institute of Technology
{zjw,kawai,yuhki}@cc.kyoto-su.ac.jp, kumamoto@net.it-chiba.ac.jp, nakajima@cse.kyoto-su.ac.jp

Abstract. Conventional pos-neg model of sentiment analysis primarily for review documents is inappropriate for news articles because of the sentiment diversity of the latter. We design three-dimension sentiments that are more suitable for the analysis of news articles. For a contentious topic, different news websites may have different sentiment tendencies and the tendencies may vary over time. To catch this feature, we construct a sentiment dictionary and develop a system that can extract news articles' sentiments, present sentiment variation over time inside a news website, and compare sentiment correlation between news websites.

1 Introduction

In recent years, the research of sentiment analysis [1] is becoming popular, which aims to automatically identify sentiments expressed in text documents. The works primarily deal with the documents that explicitly express sentiments towards a single topic, such as movie reviews and product reviews. The sentiments are usually classified to positive vs. negative categories.

In this paper, we analyze news articles both topics and sentiments of which are even more diverse than review documents. News on different topics express various kinds of sentiments, not restricted to positive-negative sentiments. For example, the reports about "sports games" are often written with "happy" or "sad" sentiments, the articles about "government's policies" mainly reflect the sentiments of "glad" or "angry," while the articles about "Libyan war" intend to convey "peaceful" or "strained" sentiments, and so on. Moreover, for some domains such as politics and economy, contentious issues continuously arise in news articles. For a contentious news topic, different news websites may have similar or opposite sentiments. Also, a news website may always persist in consistent sentiment, whereas another news website may show various sentiments over time. The extraction of these features is significant for news readers to understand the sentiment tendencies of news articles and news websites.

We first design more detailed sentiments suitable for news articles. Based on a statistical analysis and a clustering analysis, we form three-dimension sentiments: "Happy⇔Sad," "Glad⇔Angry," and "Peaceful⇔Strained." Then we

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 434–443, 2012.

construct a sentiment dictionary that stores words and their sentiment values on the three dimensions. Using the constructed sentiment dictionary, we develop a system that can detect and visualize the sentiments of news articles and news websites. The system achieves the following functions:

1. Given a news article (target article), the system can extract its topic and its sentiment.
2. The system can identify the news website (target website) of the target article, and present sentiment variation over time inside the target website related to the topic.
3. The system can calculate the sentiment correlation between the target website and other websites, and consequently extract the websites whose sentiment tendencies are similar or dissimilar to the target website.

The rest of this paper is structured as follows. Section 2 describes how to determine suitable sentiment dimensions for news articles. Section 3 describes the construction of the sentiment dictionary. Section 4 and Section 5 describe the offline and online processing of the system respectively. Section 6 evaluates the accuracy of sentiment extraction and shows the prototype of the system. Section 7 reviews related work. Finally we conclude the paper in Section 8.

2 Sentiment Dimension Design

2.1 Data Collection

From a Japanese thesaurus, we select 40 words that are thought sentiment-related, and collect evaluation data about how intensely people feel each of 40 sentiments after reading news articles. We conduct a questionnaire of 900 testees on 90 news articles. 900 testees and 90 news articles are divided equally into 9 groups, that is to say, each group consists of 100 testees and 10 news articles. Each of 100 testees in each group is asked to read each of 10 news articles in the group, and evaluate how intensely each article conveys each of 40 sentiments. The intensity scales can be selected from 5 levels: 1, 2, 3, 4, and 5, which respectively mean "strong," "relatively strong," "relatively weak," "weak," and "none." As a result, for each of 40 sentiment-related words, 9000 evaluation data (100 testees * 10 articles * 9 groups = 9000 records) are obtained.

2.2 Statistical Analysis

The means and standard deviations of 9,000 data are computed for each of 40 sentiment-related words. The results sorted in the ascending order of standard deviations are shown in Table 1. The standard deviations are between 0.84 and 1.50, and they vary substantially among 40 sentiment-related words. Since small standard deviations mean that the testees feel slightly different on the corresponding sentiments, we exclude the words "1. Modest" and "2. Sophisticated," whose standard deviations are smaller than 1.0, and their antonyms "16. Arrogant" and "7. Unsophisticated" from the list of 40 sentiment-related words.

Table 1. Means (M) and Standard Deviations (SD) for 40 sentiment-related words

ID	Sentiment-related word	M	SD	ID	Sentiment-related word	M	SD
1	Modest	4.58	0.84	...	...	...	...
2	Sophisticated	4.54	0.93	37	Serious	3.69	1.49
3	Untroubled	4.37	1.07	38	Lamentable	3.74	1.50
4	Naive	4.36	1.08	39	Awful	3.75	1.50
...	...	...	...	40	Unpleasant	3.76	1.50

2.3 Clustering Analysis

We then apply a clustering analysis to the data for the remaining 36 sentiment-related words. Each sentiment-related word is represented as a 9000-dimension vector. Hierarchical clustering method is adopted and the similarity between two words is calculated based on the Euclidean distance of two vectors. By comparing clusters at several different cluster sizes, we find the following clustering result when the cluster size is 10 is the most appropriate:

1. Happy, Bright, Refreshing, Glad, Fortunate, Favorite, Favorable, Interesting
2. Optimistic
3. Peaceful, Idyllic, Naive, Untroubled
4. Angry, Least favorite, Awful, Unpleasant
5. Sad, Dark, Brutal, Pathetic, Unfortunate, Pitiful, Serious, Lamentable, Grief-stricken
6. Strained, Fear, Sick, Uncomfortable
7. Pessimistic, Deplorable
8. Uninteresting
9. Surprising, Unexpected
10. Common

2.4 Sentiment Dimension Determination

Because there are four negative clusters (Cluster 4, 5, 6, 7) and three positive clusters (Cluster 1, 2, 3), we consider separating the sentiment-related words in the three positive clusters into four groups. In this separation, it is essential that the four newly generated positive groups correspond to the four negative clusters. In this way, four sentiment dimensions are generated. Next, the sentiment-related word "Uninteresting" representing Cluster 8 and the sentiment-related word "Interesting" from Cluster 1 pair off as the fifth dimension. Cluster 9 and Cluster 10 that are the antonyms constitute the sixth dimension. The six dimensions are as follows:

1. Happy, Bright, Refreshing
   $\Leftrightarrow$ Sad, Dark, Brutal, Pathetic, Unfortunate, Pitiful, Serious, Lamentable, Grief-stricken

Table 2. A sample of the sentiment dictionary

Word w	$s(w)$ on Happy⇔Sad	$s(w)$ on Glad⇔Angry	$s(w)$ on Peaceful⇔Strained
prize	0.862	1.000	0.808
cooking	1.000	0.653	0.881
deception	0.245	0.075	0.297
death	0.013	0.028	0.000

2. Glad, Fortunate, Favorite, Favorable
   ⇔ Angry, Least favorite, Awful, Unpleasant
3. Peaceful, Idyllic, Naive, Untroubled
   ⇔ Strained, Fear, Sick, Uncomfortable
4. Optimistic ⇔ Pessimistic, Deplorable
5. Interesting ⇔ Uninteresting
6. Surprising, Unexpected ⇔ Common

Furthermore, considering Dimension 4, Dimension 5, and Dimension 6 are strongly influenced by the background knowledge, interests, characters of news readers, we determine to use Dimension 1, Dimension 2, and Dimension 3 for news articles. By selecting the most representative word from each group, the final three sentiment dimensions are abbreviated as follows:

1. Happy ⇔ Sad
2. Glad ⇔ Angry
3. Peaceful ⇔ Strained

3 Sentiment Dictionary Construction

We then construct the sentiment dictionary, in which each entry indicates the correspondence of a target word and its sentiment values on the three dimensions. A sample of the sentiment dictionary is shown in Table 2. A sentiment value $s(w)$ of a word w on each dimension is a value between 0 and 1. The values close to 1 mean the sentiments of the words are close to "Happy," "Glad," or "Peaceful," while the values close to 0 mean the words' sentiments are close to "Sad," "Angry," or "Strained." For example, the sentiment value of the word "prize" on "Happy⇔Sad" is 0.862, which means the word "prize" conveys a "Happy" sentiment. The sentiment value of the word "deception" on "Glad⇔Angry" is 0.075, which means "deception" conveys an "Angry" sentiment.

For each of the three dimensions, we set two opposite sets (OW_L and OW_R) of original sentiment words (Table 3). The basic idea of sentiment dictionary construction is that a word expressing a left sentiment on a dimension often occurs with the dimension's OW_L, but rarely occurs with its OW_R. For example, the word "prize" expressing the sentiment "Happy" often occurs with the words "Happy," "Enjoy," "Enjoyment," "Joy," but rarely occurs with the words

Table 3. Original sentiment words for the three dimensions

Dimension	Original sentiment words
Happy ⇔Sad	Happy, Enjoy, Enjoyment, Joy (OW_L) Sad, Grieve, Sadness, Sorrow (OW_R)
Glad ⇔Angry	Glad, Delightful, Delight (OW_L) Angry, Infuriate, Rage (OW_R)
Peaceful ⇔Strained	Peaceful, Mild, Primitive, Secure (OW_L) Tense, Eerie, Worry, Fear (OW_R)

"Sad," "Grieve," "Sadness," "Sorrow." We compare the co-occurrence of each target word with the two sets of original sentiment words for each dimension by analyzing the news articles published by a Japanese newspaper YOMIURI ONLINE during 2002 - 2006.

First, for each dimension, we extract the set S of news articles including one or more original sentiment words in OW_L or OW_R. Then, for each news article, we count the numbers of the words that are included in OW_L and in OW_R. The news articles, in which there are more words included in OW_L than in OW_R, constitute the set S_L. Inversely, the news articles, in which there are more words included in OW_R than in OW_L, constitute the set S_R. N_L and N_R represent the numbers of the news articles in S_L and S_R, respectively. For each word w occurring in the set S, we count the number of news articles including w in S_L and mark it as $N_L(w)$. Similarly, we count and mark the number of news articles including w in S_R as $N_R(w)$. The conditional probabilities are

$$P_L(w) = \frac{N_L(w)}{N_L} \qquad P_R(w) = \frac{N_R(w)}{N_R}$$

A sentiment value $s(w)$ of a word w is calculated as follows:

$$s(w) = \frac{P_L(w) * weight_L}{P_L(w) * weight_L + P_R(w) * weight_R}$$

where $weight_L = log_{10}N_L$ and $weight_R = log_{10}N_R$.

4 System's Offline Processing

The system need process not only current articles but also past articles for analyzing news websites' sentiment tendencies over time. We implement a crawler for collecting news articles from 25 specified news websites (15 newspapers published in Japan and 10 newspapers' Japanese versions in other countries) every day. Then, the articles are morphologically analyzed to extract proper nouns, general nouns, adjectives, and verbs. The $tf \cdot idf$ values [2] of each extracted word in a news article are calculated.

The sentiment value of a news article is also calculated by looking up the sentiment values of the words extracted from it from the sentiment dictionary and averaging them. In this way, a news article can obtain a sentiment value ranging from 0 to 1. Considering the comprehensibility and the symmetry, the calculation value is further converted to a value ranging from -3 to 3 by the following formula: $conversion\_value = 6 * calculation\_value - 3$. When the calculation values are 1, 0.5, and 0, the corresponding conversion values become 3, 0 and -3. The conversion values 3, 2, 1, 0, -1, -2, -3 on a dimension, e.g., "Happy⇔Sad," correspond to "Happy," "Relatively happy," "A little happy," "Neutral," "A little sad," "Relatively sad" and "Sad" respectively.

The above processing is done offline. As a result, the collected news articles, the $tf \cdot idf$ values of the words extracted from the news articles, and the sentiment values of the news articles are stored in a database.

5 System's Online Processing

5.1 Extracting the Topic and the Sentiment of the Target Article

Given a news article, the system first extracts keywords and sub-keywords representing the article's topic. The keywords are the top five words with the highest $tf \cdot idf$ values extracted from the target article. The sub-keywords are the top five words with the highest sums of $tf \cdot idf$ values in the related articles that include any of the five keywords. Both the five keywords and the five sub-keywords are presented to the user. The user selects the words representing the topic that he or she has concern about. The selected words are later used to retrieve past articles for analyzing sentiment tendencies of news websites related to the concerned topic. The sentiment value of the target article is also calculated by using the sentiment dictionary and converted to a conversion value. The conversion values of the target article on the three dimensions are also presented to the user. An example is shown in Figure 6 of Section 6.2.

5.2 Presenting Sentiment Variation Inside the Target Website

The news website of the target article is identified by analyzing the URL of the article. Figure 1 is an example of the sentiment variation over time inside the target website. The news articles including the user-selected words in the target website date back at a regular interval t_i (e.g., one day or two days). At each interval t_i, the articles, on which the $tf \cdot idf$ values of the user-selected words are larger than a threshold τ_0, are extracted. Their sentiment values are calculated, converted and averaged as the sentiment values $s(t_i)$ of the target website at the interval t_i. The real horizontal line represents the mean of sentiment values and the dotted horizontal lines represent the standard deviation of sentiment values. By browsing the sentiment variation inside the target website, users can perceive whether the sentiment of the current article (the red point) is consistent with the past sentiments of the target website. An example is shown in Figure 7 of Section 6.2.

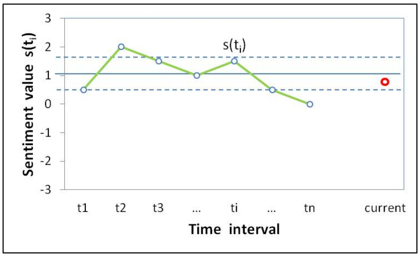


Fig. 1. Comparison inside a website

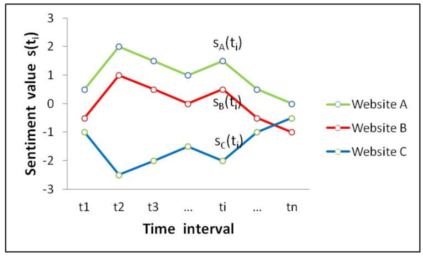


Fig. 2. Comparison between websites

5.3 Presenting Sentiment Correlation between Websites

Another function of the system is to show the correlation of sentiment tendencies between the target website and its counterpart websites. We calculate the correlation coefficient $\rho(X, Y)$ for two websites X and Y. Let $s_X(t_i)$ and $s_Y(t_i)$ be the sentiment values of two websites X and Y at the interval t_i respectively, and we calculate their correlation coefficient $\rho(X, Y)$ as follows:

$$\rho(X,Y) = \frac{\sum_{i=1}^{n}(s_X(t_i) - \overline{s_X}) * (s_Y(t_i) - \overline{s_Y})}{\sqrt{\sum_{i=1}^{n}(s_X(t_i) - \overline{s_X})^2} * \sqrt{\sum_{i=1}^{n}(s_Y(t_i) - \overline{s_Y})^2}}$$

where $\overline{s_X}$ and $\overline{s_Y}$ are the means of $s_X(t_i)$ and $s_Y(t_i)$ during the analysis period. Figure 2 shows an extreme example, in which $\rho(A, B)$ is 1 (direct correlation) and $\rho(A, C)$ is -1 (inverse correlation). Based on the calculation results of correlation coefficient, the system can extract sentiment-similar websites by selecting the ones, ρ between which and the target website is larger than a threshold τ_1 (e.g., 0.5), and sentiment-dissimilar websites by selecting the ones, ρ between which and the target website is smaller than another threshold τ_2 (e.g., -0.5). An example is shown in Figure 8 of Section 6.2.

6 Evaluation and Observation

6.1 Evaluation on Sentiment Extraction Accuracy

We specify 10 news domains (Society, Sports, Economy, Synthesis, Politics, Overseas, Life, Entertainment, Culture, and Science) and pick up 10 news articles from each domain. As a result, 100 news articles are selected for evaluating sentiment extraction accuracy. 5 testees are asked to read each of 100 news articles and evaluate how intensely they feel the sentiments on the three dimensions. Each testee can evaluate the sentiment intensity by giving an integer from -3 to 3. For example, for the dimension "Happy⇔Sad," 3, 2, 1, 0, -1, -2, -3 represent "Happy," "Relatively happy," "A little happy," "Neutral," "A little sad," "Relatively sad," and "Sad" respectively. The evaluation values from 5 testees for each article and each dimension are averaged as the mean value of the article's

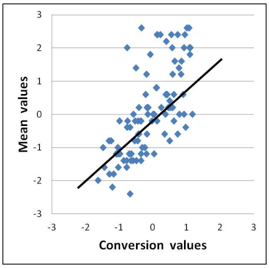


Fig. 3. Happy⇔Sad

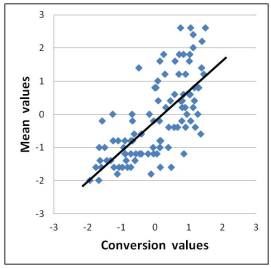


Fig. 4. Glad⇔Angry

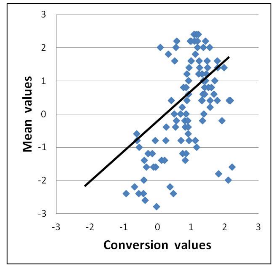


Fig. 5. Peaceful⇔Strained

sentiment on that dimension. For each of 100 news articles, the conversion value of the sentiment on each dimension is also calculated by using the sentiment dictionary and the conversion formula.

We compare the conversion values (computer's output) of 100 articles with the mean values (testees' evaluation). The results on the three dimensions are shown in Figure 3, Figure 4, and Figure 5, respectively. An article is presented by a point in the graph, the abscissa of which is its conversion value and the ordinate of which is its mean value. Most of the points are near to the diagonal, which means the conversion values approximate to the mean values. The average errors between the conversion values and the mean values on "Happy⇔Sad," "Glad⇔Angry," and "Peaceful⇔Strained" are 0.748, 0.746, and 1.128, respectively. Considering the sentiment values have seven levels ranging from -3 to 3, the error of about one level on each dimension indicates that our sentiment extraction accuracy is good.

6.2 Observation on Sentiment Tendencies

We implement a prototype that extracts the sentiment of a news article, the sentiment variation inside a website, and the sentiment tendencies' correlation between different websites. An example is that a user is browsing a news article reporting the draft of tax increase for the revival of Japanese earthquake. The system first extracts the keywords and the sub-keywords representing the topic and the sentiment of the news article (Figure 6).

After the user selects the concerned words (e.g., "tax increase" and "revival"), the system identifies the website of the article is "NIKKEI" and analyzes the sentiment variation related to "tax increase for the revival" inside the website "NIKKEI" (Figure 7). The overall sentiment tendency about the topic in this website is relatively "Sad" and the sentiment of the current target article (the red point) keeps within the sentiment variation range of the website.

Figure 8 shows the system detects the sentiment-similar website "Asahi" and the sentiment-dissimilar website "Mainichi" on "Happy⇔Sad" related to the topic "tax increase for the revival" for the target website "NIKKEI." The graph shows the comparison results of sentiment correlation between those websites. The green

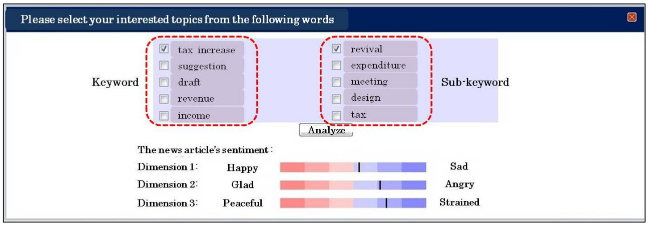


Fig. 6. Snapshot of topic and sentiment extraction

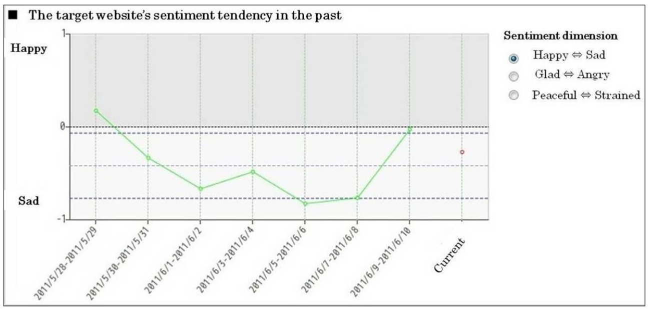


Fig. 7. Snapshot of sentiment variation inside a website

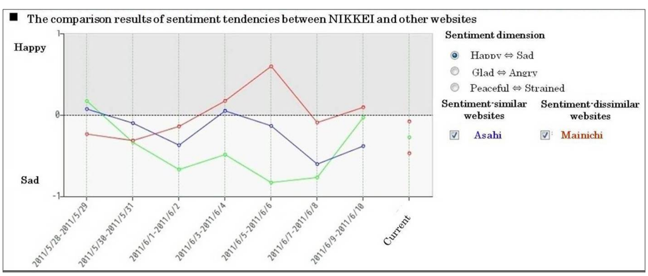


Fig. 8. Snapshot of sentiment correlation between websites

line represents the sentiment tendency of the target website "NIKKEI," the blue line represents the sentiment tendency of its sentiment-similar website "Asahi," and the red line represents the sentiment tendency of its sentiment-dissimilar website "Mainichi." From the graph, the user can exactly observe that "Asahi" has similar tendencies to "NIKKEI" while "Mainichi" has opposite tendencies to "NIKKEI."

7 Related Work

Sentiment analysis is increasingly important in many research areas. Turney [3] proposed a method for classifying reviews into two categories: recommended and not recommended based on mutual information. Pang et al. [4] extracted the subjective portions of movie reviews and classified them as “thumbs up” or “thumbs down” by applying text-categorization techniques. These methods only consider positive-negative sentiment. Unlike these methods, our system captures more detailed sentiments of three dimensions suitable for news articles.

There also exist other several sentiment models. Plutchik [5] designed a four-dimension model. Russell [6] proposed a two-dimensional space. Pitel et al. [7] considered 44 paired emotion directions and created a French sentiment dictionary using a SVM classifier. Different from their works, we adopt three-dimension sentiments for news articles by conducting a statistical analysis and a clustering analysis of 40 sentiment-related words.

8 Conclusion

We designed three-dimension sentiments for news articles, and developed a system for extracting a news article’s sentiment, finding the sentiment variation inside a news website, and comparing sentiment tendencies between different websites. Our implementation enabled users to obtain visual comparison results.

Acknowledgments. This work was supported in part by SCOPE (Ministry of Internal Affairs and Communications, Japan), by the MEXT Grant-in-Aid for Young Scientists(B) (#21700120), and by the MEXT Grant-in-Aid for Scientific Research(C) (#23500140).

References

1. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval 2(1-2), 1–135 (2007)
2. Ricardo, B.Y., Berthier, R.N.: Modern Information Retrieval: The Concepts and Technology behind Search, 2nd edn. Addison-Wesley Professional (2011)
3. Turney, P.D.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In: ACL 2002, pp. 417–424 (2002)
4. Pang, B., Lee, L.: A Sentiment Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In: ACL 2004, pp. 271–278 (2004)
5. Plutchik, R.: The Emotions. Univ. Pr. of Amer. (1991)
6. Russell, J.A.: A Circumplex Model of Affect. Journal of Personality and Social Psychology 39(6), 1161–1178 (1980)
7. Pitel, G., Grefenstette, G.: Semi-automatic Building Method for a Multidimensional Affect Dictionary for a New Language. In: LREC 2008 (2008)

Evolution of Extensible Java EE-Based Agent Framework

Mirjana Ivanović[1], Milan Vidaković[2], Dejan Mitrović[1], and Zoran Budimac[1]

[1] Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Serbia
{mira,dejan,zjb}@dmi.uns.ac.rs
[2] Faculty of Technical Sciences, University of Novi Sad, Serbia
minja@uns.ac.rs

Abstract. In this paper the evolution of *EXtensible Java EE-based Agent Framework* (*XJAF*), a *FIPA*-compliant multi-agent system will be presented. The main motivation behind the development of *XJAF* is to demonstrate how existing, standardized *Java EE* technologies, libraries and technical solutions (e.g. *JNDI*, *JMS*, and *EJB*), can be used to implement a large subset of functionalities required by a multi-agent system. Immediate benefits of this approach are shorter development time, harnessing of advanced programming features such as agent pooling and runtime load-balancing, flatter learning curve for new developers of the system, and so on. Several weaknesses that became apparent with the original implementation of *XJAF* caused recent improvements of the system. These include the addition of fault-tolerant techniques, and the increase of interoperability through a *SOA*-based design.

Keywords: multi-agent systems, agent development.

1 Introduction

By analyzing multi-agent systems (*MAS*) that have been developed during the last decade [3], it can be concluded that in the majority of cases many functional requirements were implemented from scratch. This, however, could have been avoided, since Java EE already includes many technical solutions suitable for implementing a large subset of functional requirements imposed on MAS's.

Several years ago we have developed a *FIPA*-compliant [9] multi-agent system named *EXtensible Java EE-based Agent Framework* (*XJAF*) [21, 22, 23]. The main motivation was to demonstrate how the agent technology can benefit from existing, standardized *Java EE*-based software development technologies. Benefits include shorter development time, standards compliance, harnessing of advanced programming features such as agent pooling and runtime load-balancing, flatter learning curve for new developers of the system, and so on.

Over the time, *XJAF* has been successfully applied to two real-life applications in the field of distributed digital library catalogues [21, 22]. But, in spite of the

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 444–453, 2012.

successful usage, and along with further developments in the domain of agent technology, the system has recently been critically evaluated and improved. The overall error resilience of the system has been improved with the introduction of fault-tolerant techniques [15, 16]. Additionally, components of the old system have been redesigned as web services, which improved their interoperability [17].

The rest of the paper is organized as follows. Section 2 brings a comparison of existing *MAS* architectures with *XJAF*. Section 3 provides information on the system architecture, both the old and the new. A simple example of using *XJAF* is presented in Section 4. Recent changes and improvements of the system are presented in Section 5. Finally, Section 6 outlines general conclusions and future work.

2 Related Work

During the last decade, many authors have made different design and implementation choices during the development of their *MAS's*. A representative subset of these solutions includes *ABLE, Aglets, DimaX, FUSION@, JADE,* and *Voyager*.

Agent Building and Learning Environment (*ABLE*) [1, 5, 14] is a Java-based, *FIPA*-compliant *MAS* and a framework for agent development based on machine learning and reasoning. It uses the *JavaBeans* technology for building both agents (*AbleAgents*) and general-purpose, reusable software components (*AbleBeans*). *ABLE* supports rule-based reasoning and includes a number of inference engines, based on backward and forward chaining, fuzzy logic, neural networks, scripting, and more.

Aglets [2, 13, 18] is a Java-based *MAS* with the support for agent mobility, synchronous and asynchronous message exchange, and security mechanisms. It is a multi-tier environment, incorporating following tiers: *AgletsRuntime* – a host for agents, proxies, and contexts, *Agent Transport and Communication Interface* (*ATCI*) – a high-level communication and mobility infrastructure, and *Agent Transport Protocol* (*ATP*) – a low-level infrastructure for sending agents and messages across *TCP/IP* networks.

DimaX [7, 11] is a *MAS* built around fault-tolerance concepts, including fault detection and monitoring, and adaptive agent replication. Fault detection is based on the *heartbeat* technique, in which agents ping each other in order to indicate their availability. The monitoring service tries to predict future failures of the system or individual agents. For adaptive replication, *DimaX* maintains *interdependency graphs* in order to identify more important agents, and adapt the number of replicas accordingly.

FUSION@ [19, 20] is a modular, *SOA*-based *MAS* with a layered organization. At the lower level, services are used to implement functionalities of the framework. They can be invoked locally or remotely, and can be organized as local or web services. At the upper layer, deliberate, *BDI*-style agents control task distribution: they employ load-balancing and quality assurance analysis in order to assign the given task to the most appropriate service.

Java Agent DEvelopment Framework (*JADE*) [4, 12] is probably the most widely-used *FIPA*-compliant *MAS* that utilizes advanced Java features for its functioning. It has a built-in support for agent mobility and includes various security features: authentication, authorization, and encryption of exchanged *FIPA ACL* messages. One of the most important features is its extensibility through third-party add-ons.

Voyager [24] is primarily an application server, but it can also be used as a multi-agent system. Its agents interact by exchanging messages using several communication patterns, such as synchronous (blocking), one-way, and one-way multi-cast. Agent mobility is also supported, and relies on *Java Serialization*.

The main difference between *XJAF* and presented *MAS's* is in the use of *Java EE* as the implementation platform. *Java EE* includes technologies, libraries and tools that simplify the development of secure, efficient, reliable, and scalable software. This makes it an excellent platform for *MAS* development. For example, by wrapping agents inside *EJB* components, *XJAF* delegates the agent life-cycle management to the underlying enterprise application server. This approach increases the system's performance, because modern application servers employ *EJB* pooling and runtime load-balancing techniques. Out of the *MAS's* described above, only *Voyager* attempts to fully utilize the *Java EE* platform. However, *Voyager* is an application server written from scratch, and a commercial product. *XJAF*, on the other hand, executes inside *GlassFish* [10], an existing, open-source solution. *BlueJADE* [6] is a project aimed at incorporating *Java EE* with *JADE*, but it only transforms the entire framework into a manageable service.

In a recent improvement of the *XJAF*, a special care has been dedicated to building and maintaining fault-tolerant networks (described in more details in Section 5). None of these systems have taken the fault-tolerant network management to this extent.

Out of the presented systems, only *FUSION@* uses web services as building blocks, while *JADE* relies on an add-on to employ web services for interoperability. The service-oriented design is at the core of a recently improved version of *XJAF*.

3 The *XJAF* Architecture

XJAF is designed as pluggable software architecture with set of loosely-coupled, dynamically loaded modules called *managers*. A manager is responsible for handling a distinct part of the overall agent-management process. Managers are recognized and used solely by their interfaces, which allows for custom implementations. Also, the set of managers is not hard-coded: new managers (that is, new functionalities) can be added as needed, requiring no changes to the underlying implementation.

Managers of *XJAF* are controlled by *Facilitator*, the central component of the system. Besides serving as a connection point for the system's managers, Facilitator provides a working environment for *XJAF* agents, through an exposed *API*.

Agent management – *AgentManager* includes functionalities that correspond to *FIPA's Agent Directory Service* [9]. Its main tasks are to keep a directory of agents, and to activate and deactivate agents on demand. For agent directory services, *AgentManager* uses *Java Naming and Directory Interface*. To support agent mobility, it actually keeps two directories: one for agents available locally, and another for agents that have left this instance and moved to a remote *XJAF*. Later, the agent can easily be located either by extracting a record from the first directory, or by forwarding the request to the remote *XJAF* that has been recorded in the second directory.

Task management – Capabilities of *XJAF* agents are described in form of *tasks*. Each agent publishes a set of tasks it can perform into a centralized repository handled by *TaskManager*. For interoperability reasons, tasks are represented and stored using standardized and widely-used *W3C XML Schema* language [25]. A client (either another agent or an external user) can request task execution by sending a message to *TaskManager*. In response, the manager finds and employs the best-suited agent, and later returns the result to the client. Internally, agents exchange task requests and execution results using the messaging infrastructure of *XJAF*. For external clients, the system uses the concept of event listeners, described later.

Agent communication – In *XJAF*, *MessageManager* is the module in charge of inter-agent communication. It utilizes *Java Message Service* (JMS) for the actual message exchange, and supports three communication patterns:

- Agent ↔ *XJAF*.
- Agent ↔ Agent, where both agents are located in the same *XJAF*.
- Agent ↔ Agent, where agents are distributed over a network.

Originally, *XJAF* used *KQML* [8] as the message format. Over the time, however, *FIPA ACL* [9] has emerged as a new agent communication standard. In order to increase the interoperability of *XJAF* agents, the work is underway to update the existing messaging infrastructure of *XJAF* and replace *KQML* with *FIPA ACL*.

To simplify the agent development process, *XJAF* offers the so-called *programmatic mode*. The basic idea is to replace the process of constructing a message, sending it, and waiting for the reply with a single method invocation. *XJAF* offers a number of methods that correspond to *KQML* performatives, and developers are free to choose between the programmatic mode and the plain messaging system.

Connecting distributed *XJAF* instances – In addition to the functionalities offered by *AgentManager* that serve as a support for agent mobility, *XJAF* includes a framework for interconnecting distributed instances of the environment. The main motivation behind the network organization of *XJAF* instances is runtime load-balancing. That is, instead of running a large number of agents on a single machine, it is more efficient to use several computers, and then distribute agents among them.

In the original implementation, distributed *XJAF* instances were organized into a tree-like structure, with the primary *XJAF* representing the root of the tree. Recently, the tree-like structure has been replaced by a fully-connected graph, in an effort to increase the fault-tolerance of XJAF networks (as described in Section 5).

Security features of *XJAF* – *XJAF* supports a variety of security mechanisms: protection of agents and the system from malicious attacks, code and data integrity, and the integrity and confidentiality of exchanged messages. As security mechanisms can impose significant overhead to the code execution, they are not applied automatically, but are available as methods of a distinct manager named *SecurityManager*. The choice of techniques and algorithms to be used for implementing methods of *SecurityManager* is left to the system's developer. The default implementation uses *Java Public Key Infrastructure* (PKI).

Service manager – In *XJAF*, a *service* is a reusable software component that implements some specific functionality. It can be used by agents and other entities of the system. The basic idea is to extract some common tasks (such as input validation) into services, and avoid repeating the implementation for each agent. *ServiceManager* offers methods for adding, removing, searching, and using services. Similarly to *AgentManager*, *ServiceManager* uses *JNDI* for service directory implementation. Also, *XJAF* services are, as agents, *POJOs* embedded inside *EJB* components.

4 Example: *Factorial* Agent

XJAF has been applied in the field of distributed digital library catalogues: the *virtual central catalogue* and *metadata harvesting* [22, 21]. Two fundamental features employed in these applications are mobility and agents' inter-relationship. In the virtual central catalogue, agents migrate from one node to another in search of the content specified by the query issued from the central node. In case of metadata harvesting, central agent delegates tasks to agents in subordinate nodes and collects results.

In both cases, the use of agent technology provided a simple and effective mechanism for a dynamic setup of the distributed catalogues network. The main contribution of this approach is a solution that gives a simple and flexible mechanism for automatic maintenance of dynamically changing networks, as well as an environment suitable for the implementation of additional services in distributed libraries.

In order to demonstrate the usage of *XJAF*, a simple framework for calculating the factorial is developed. Although the presented example is artificial and simple, it is adequate for demonstrating the fundamentals of writing and employing *XJAF* agents, and for gaining a deeper understanding of the system's functioning.

The framework consists of *FactorialAgent*, an agent that perform the actual calculation, and an *XML Schema*-based task description. The agent follows a simple set of rules: for $n \geq 2$ it asks another agent to calculate $(n - 1)!$, and then returns the result by multiplying n with the received sub-result. The implementation of this algorithm is given next. The *FactorialTask* class is generated from the defined task description, using *Java Architecture for XML Binding* (*JAXB*).

```
public AgentResult execute(AgentTask task, Facilitator f, Object aId) {
  if (task instanceof FactorialTask) {
    FactorialTask factTask = (FactorialTask)task;
    int value = factTask.getInputValue();
    if (value >= 2) { // ask an agent to calculate (n-1)!
      Object newId = f.recommendOne(task);
      factTask.setInputValue(value - 1);
      AgentResult sub = facilitator.execute(newId, task);
      // return the final result
      value *= subResult.getContent();
      return new AgentResult(value); }
    return new AgentResult(1); } // 1!, 0!
  return null; } // unsupported task
```

External clients interact with agents in *XJAF* through the *FacilitatorProxy* component. This component is an abstraction over *Facilitator*, and hides all the “low-level” details, such as *JNDI* lookup and *JMS* message composition. To interact with an agent, the client simply needs to create an instance of *FacilitatorProxy* and pass it the task object along with concrete parameter values.

The source code of the external client, named *FactorialClient*, is shown next. Its method *calculate()* receives the value for *n* and generates an appropriate *FactorialTask* object, and then passes it on to *FacilitatorProxy*. The last parameter of the *FacilitatorProxy’s execute()* method is an instance of the class implementing the *AgentListener* interface. This interface is used for event listener-based communication between *XJAF* and external clients. For example, once the task is completed, the listener’s *actionPerformed()* method will be called.

```
class FactorialClient implements AgentListener {
  public void calculate(int n) {
    FactorialTask task = new FactorialTask();
    task.setInputValue(n);
    // request task execution
    FacilitatorProxy fp = new FacilitatorProxy();
    fp.execute(task, this); }
  // methods of the AgentListener interface
  public void actionStarted(AgentEvent e) { }
  public void actionPerforming(AgentEvent e) { }
  public void actionPerformed(AgentEvent e) { } }
```

5 Recent Improvements of the Architecture

After the *XJAF* system was put into practice, several weaknesses of the original architecture were recognized. First of all, it did not include any techniques that would allow the network of distributed *XJAF* instances to recover from faulty situations. Secondly, there was an overall lack of interoperability in the sense that only Java-based external clients were able to use *XJAF*. Improvements to the original *XJAF* architecture that incorporate solutions to these problems are described next.

Fault-tolerant networks of *XJAF* instances – Although it was easy to build, the original tree-like structure of interconnected *XJAF* instances was very fragile. That is, failure of any instance would divide the tree into two sets of mutually unreachable instances. In order to increase the fault-tolerance of *XJAF* networks, the tree-like structure was replaced by a fully-connected graph [15, 16]. Fully-connected graphs are convenient because, for a mobile agent, the problem of finding a path to the target *XJAF* becomes trivial. The increase in time needed to add a new instance to an existing network is managed by exploiting the mobility feature of agents.

The improved version of *XJAF* incorporates a specialized, light-weight mobile agent - *ConnectionAgent*. The initial job of the agent is to register its newly created host *XJAF* with an existing network. Afterwards, the agent performs tasks related to detecting a broken neighbor and informing other instances of the problem.

ConnectionAgent carries with itself a set of all *XJAF* instances it has detected so far. An element of this set is a triplet (*address, state, timestamp*), where *address* is the instance's network address, *state* is its running state, as perceived by the agent (e.g. *running*, *unresponsive*, etc.), while *timestamp* marks the time at which the perceived state was reached. Timestamps are crucial tools for both registration and failure detection processes, and are used to distinguish between mutually exclusive messages.

An *XJAF* instance that wants to connect to an existing network initiates and dispatches its own *ConnectionAgent*. When the agent visits an existing *XJAF* instance in the network, it synchronizes the set of triplets with its current host. During the synchronization, the hosting *XJAF* will learn about the agent's home. At the same time, the agent will learn about all instances its current host is or was connected to. Any changes made to the agent's set are sent back home.

After the synchronization procedure, the agent divides its set of yet unvisited instances into two distinct subsets. It then spawns a clone of itself and assigns one of these subsets to it, while keeping the other for itself. Both agents migrate to the next unvisited host from their own subsets, repeating the synchronization procedure, and re-spawning new *ConnectionAgents* at each subsequently visited *XJAF*. The registration procedure is finished for an agent once it has no more unvisited instances.

By performing the set division and agent re-spawning at each visited host, the overall dependence of the registration time on the number of existing hosts becomes insignificant [15]. Once the registration procedure is completed, all *ConnectionAgents* return to their home *XJAF*. One of the instances is selected to establish a *heartbeat* connection with its neighbors in order to perform failure detection. An agent that detects a failure of a neighbor will inform all other instances of the problem. The heartbeat connection is made in a way that assures that all remaining instances have a correct view of the network state, regardless of the number of failed instances [15].

SOM*: *SOA*-based *MAS – A major disadvantage of the original *XJAF* architecture was the lack of interoperability: only Java-based external clients could use the system and interact with its agents. In order to overcome this obstacle and encourage its wide-spread use, the managers of the original *XJAF* have been re-implemented as web services. The most important improvement brought by the new system, named *SOA-based MAS* (*SOM*) [17], is increased interoperability and standards compliance. That is, unlike with *XJAF*, any external entity that supports *SOAP*, the standardized communication protocol, can act as a client of *SOM*.

Because the client and *SOM* can be implemented using different programming languages and platforms, it is important to define a proper way of encoding/decoding task execution requests and results inside *SOAP* envelopes. Since the *SOM Facilitator* is a web service, it has a strict method signature convention. At the same time, however, the system should be flexible enough for describing any task. Therefore, in *SOM*, *XML* has been selected as the task description language.

When making a request for task execution, the client describes the request using *XML*. The *XML* description itself must be in compliance with an appropriate *XML Schema* document, which is deployed on the target *SOM*, and is used to validate the request. The *XML* descriptor must also include the appropriate namespace identifier in order to create a mapping between the task and an appropriate agent.

The following listing outlines the *FactorialClient* example shown earlier, but rewritten for *SOM*. It offers two major improvements over the previous approach:

1. There is a greater decoupling between the client and the agent framework, because the client does not need a reference Java implementation of the task (in the previous client example, the *FactorialTask* class).
2. The client no longer uses a custom, *AgentListener* interface, but rather relies on the standardized Java framework for asynchronous invocations of web services.

```
class FactorialClient implements AsyncHandler<ExecuteResponse> {
  public void calculate(int n) {
    String task = "<?xml version=\"1.0\" encoding=\"UTF-8\"?>" +
      "<p:factTask xmlns:p=\"xjafs/tasks/factTask\"" +
      "xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"" +
      "xsi:schemaLocation=\"xjafs/tasks/factTask FactTask.xsd\">" +
      "<p:inputValue>10</p:inputValue> </p:factTask>";
    // request task execution
    FacilitatorProxyService fps = new FacilitatorProxyService();
    FacilitatorI fac = fps.getFacilitatorProxyPort().getFacilitator();
    fac.executeAsync(task, this); }
  // asynchronous result handler
  @Override public void handleResponse(Response<ExecuteResponse> res) {}
```

SOM is a conceptual specification of web services, and can be implemented using many modern programming languages. The default implementation is provided in *Java EE*, following the same idea behind *XJAF* – reuse of existing technologies for MAS development as much as possible. However, this concept poses a major problem: an agent written for a *Java EE*-based implementation of *SOM* cannot migrate to, for example, a *Python*-based implementation. Because it severely undermines the design goal of *SOM*, this problem needed to be alleviated.

In order to overcome this issue, a new agent-oriented programming language - *Agent LAnguage for SOM* (*ALAS*) – has been proposed [17]. Besides providing developers with programming constructs that hide the overall complexity of agent development, one of the main goals of *ALAS* is to hide the implementation details of underlying *SOM* instances. It enables the developers to focus on concrete problem solving, and not have to worry about the *SOM* implementation language. *ALAS* compiler includes a *hot compilation* feature: once a mobile, *ALAS*-based agent arrives to a *SOM* instance, its executable code will be regenerated on-the-fly to match the underlying implementation platform.

The following listing outlines the *FactorialAgent* example shown earlier, in Section 4, but re-implemented using *ALAS*. The advantages of developing agents using *ALAS* in *SOM* over pure Java in *XJAF* are obvious. The agent source code is much shorter and cleaner, and the overall process is more “natural” and in accordance to the agent-oriented development philosophy (e.g. there are no specific interfaces to implement and no unnecessary type-castings or data extractions to be performed).

```
agent FactorialAgent {
  service int calculate(int n) {
    if (n < 2) return 1; // 1!, 0!
    AID aid = facilitator.newAgent("FactorialAgent");
    int sub = sendMsg(aid, "calculate", "n" = (n - 1));
    return n * sub; } }
```

6 Conclusion and Future Work

The main advantage of *XJAF* over many other existing *MAS's* is the full employment of *Java EE* technologies, tools, and libraries that simplify the development of robust, reliable, and scalable software.

The interoperability of *XJAF* has recently been increased significantly by following the *SOA* philosophy. The *SOA*-based design enables any external client to use *SOM* and interact with its agents

The employment of web services is not enough for achieving the full interoperability. The new agent-oriented programming language *ALAS* and its supporting tools provide an opportunity for agent mobility in, platform-wise, heterogeneous networks. The execution code of an agent written in *ALAS* can be regenerated at runtime, during the migration process, to match the implementation platform of the new host.

Additionally, the original *XJAF* has recently been improved with the introduction of fault-tolerant techniques. The introduced robust agent tracking technique can locate a mobile agent even if a number of *XJAF* instances in its path break.

The future work on *SOM* will be concentrated on improving its interoperability and fault-tolerance even further. The focus of further development will also be on *ALAS*, in enriching the language with new programming constructs, improving its performance, and adding support for the development of *BDI*-style agents.

Acknowledgements. Research presented in this paper was supported by Ministry of Science and Technological Development of Republic of Serbia, Grant III-44010, Title: Intelligent Systems for Software Product Development and Business Support based on Models.

References

1. ABLE homepage, `http://www.alphaworks.ibm.com/tech/able`
2. Aglets homepage, `http://aglets.sourceforge.net`
3. Bădică, C., Budimac, Z., Burkhard, H., Ivanović, M.: Software agents: languages, tools, platforms. Computer Science and Information Systems, ComSIS 8(2), 255–296 (2011)
4. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing multi-agent systems with JADE. John Wiley and Sons (2007)
5. Bigus, J.P., Schlosnagle, D.A., Pilgrim, J.R., Mills, W.N., Diao III, Y.: ABLE: a toolkit for building multiagent autonomic systems. IBM Systems Journal 41(3), 350–371 (2002)
6. Cowan, D., Griss, M., Burg, B.: BlueJADE – a service for managing software agents. Technical report. Hewlett-Packard Company (2002)

7. Faci, N., Guessoum, Z., Marin, O.: DimaX: a fault-tolerant multi-agent platform. In: Proceedings of the 2006 International Workshop on Software Engineering for Large-Scale Multi-Agent Systems, pp. 13–20 (2006)
8. Finin, T., Fritzson, R., McKay, D., McEntire, R.: KQML as an agent communication language. In: Proceedings of the Third International Conference on Information and Knowledge Management, CIKM 1994, New York, NY, USA, pp. 456–463 (1994)
9. FIPA homepage, http://www.fipa.org/
10. GlassFish open source application server homepage, http://glassfish.java.net/
11. Guessoum, Z., Faci, N., Briot, J.P.: Adaptive replication of large-scale multi-agent systems: towards a fault-tolerant multi-agent platform. In: Proceedings of the Fourth International Workshop on Software Engineering for large-Scale Multi-Agent Systems, pp. 1–6 (2005), http://www.fipa.org/specs/fipa00001/SC00001L.pdf (2002), http://www.fipa.org/specs/fipa00061/SC00061G.pdf (2002)
12. JADE homepage, http://jade.tilab.com/
13. Karjoth, G., Lange, D.B., Oshima, M.: A security model for Aglets. IEEE Internet Computing 1(4) (1997)
14. Meyer, M.: The features and facets of the Agent Building and Learning Environment, http://www.ibm.com/developerworks/autonomic/library/ac-able1/
15. Mitrović, D., Budimac, Z., Ivanović, M., Vidaković, M.: Agent-based approaches to managing fault-tolerant networks of distributed multi-agent systems. Multiagent and Grid Systems 7(6), 203–218 (2011), doi:10.3233/MGS-2011-0176
16. Mitrović, D., Budimac, Z., Ivanović, M., Vidaković, M.: Improving fault-tolerance of distributed multi-agent systems with mobile network-management agents. In: Proceedings of the International Multiconference on Computer Science and Information Technology, vol. 5, pp. 217–222 (2010)
17. Mitrović, D., Ivanović, M., Vidaković, M.: Introducing ALAS: a novel agent-oriented programming language. In: Symposium on Computer Languages, Implementations, and Tools (SCLIT), International Conference on Numerical Analysis and Applied Mathematics (ICNAAM), AIP Conf. Proc., Halkidiki, Greece, vol. 1389, pp. 861–864 (2011)
18. Tai, H., Kosaka, K.: The Aglets project. Communications of the ACM 42(3), 100–101 (1999)
19. Tapia, D.I., Bajo, J., Corchado, J.M.: Distributing Functionalities in a SOA-Based Multi-agent Architecture. In: Demazeau, Y., Pavón, J., Corchado, J.M., Bajo, J. (eds.) 7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009). AISC, vol. 55, pp. 20–29. Springer, Heidelberg (2009)
20. Tapia, D.I., Rodrıguez, S., Bajo, J., Corchado, J.M.: FUSION@, a SOA-Based Multi-Agent Architecture. In: Corchado, J.M., et al. (eds.) DCAI 2008. ASC, vol. 50, pp. 99–107. Springer, Heidelberg (2009)
21. Vidaković, M.: Extensible Java based agent framework. PhD thesis, Faculty of Technical Sciences, University of Novi Sad, Serbia (2003)
22. Vidaković, M., Milosavljević, B., Konjović, Z., Sladić, G.: EXtensible Java EE-based agent framework and its application on distributed library catalogues. Computer Science and Information Systems, ComSIS 6(2), 1–16 (2009)
23. Vidaković, M., Sladić, G., Konjović, Z.: Security management in J2EE based intelligent agent framework. In: Proceedings of the 7th IASTED International Conference on Software Engineering and Applications (SEA 2003), pp. 128–133 (2003)
24. Voyager homepage, http://www.recursionsw.com/Products/voyager.html
25. World Wide Web Consortium (W3C) XML Schema, http://www.w3.org/XML/Schema

Forecasting Consumer Interest in New Services Using Semantic-Aware Prediction Model: The Case of YouTube Clip Popularity

Luka Vrdoljak[1], Vedran Podobnik[2], and Gordan Jezic[2]

[1] Erste & Steiermärkische Bank, Croatia
lvrdoljak@erstebank.com
[2] University of Zagreb, Faculty of Electrical Engineering and Computing, Croatia
{vedran.podobnik,gordan.jezic}@fer.hr

Abstract. With intense increase in number of competing service providers in the information and communication sector, companies must implement mechanisms for forecasting consumer interest in new services. Common growth models provide the mechanisms for modelling and predicting acceptance of a certain service. However, they have two shortcomings: i) limited precision; and ii) a short, but yet existing, time delay. By using semantic reasoning for detecting similarities between services already on a market and ones that are just to be introduced, it is possible both to increase forecasting precision and eliminate the time delay caused by the need to collect a certain amount of data about the new service before a prediction can be made. The proposed semantic-aware prediction model is elaborated on a case of forecasting YouTube clip popularity.

Keywords: Consumer Relationship Management, Consumer Managed Relationship, Forecasting, Growth Models, Semantic Reasoning, YouTube.

1 Introduction

Considering the increasing number of competing service providers and services on the information and communication market, companies must focus on maintaining consumer satisfaction. In order to do so, service providers must observe their consumers individually, rather than seeing them just as a part of a certain market niche. Such individual and personalized approach can be recognized on the market through two most common concepts: *Consumer Relationship Management* (CRM) and *Consumer Managed Relationship* (CMR) [1][2]. These two concepts reside on three basic ideas: i) consumer experience management (CEM), ii) real-time analysis, and iii) technology used for cost decreasing and creating a consumer-oriented environment [3][4][5].

Most companies react only when a number of consumers decreases (i.e. churn rate dominates over growth rate). However, by then it is usually too late to intervene. On the other hand, using *Predictive Analysis* (PA) can lead to a proactive consumer retention strategy [4]. By analysing consumer habits, expenditure and other behaviour

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 454–463, 2012.

patterns, forecasting models can determine the probability of a decrease in consumers' interest in a certain service, or even the potential interest in a service that has yet to be introduced in the market [6]. Additionally, predictions could be improved with the use of Semantic Web technologies which enable detection of similar services already on the market.

In this paper, we will first make an insight in the common forecasting models and state-of-the-art technologies for semantic service profiling (Section 2). Then we will present our proposed system through key processes (Section 3) and its architecture (Section 4). In Section 5 we will conduct an evaluation of our ideas on YouTube clip popularity forecast. Section 6 concludes the paper and presents the future research planned to result in original scientific contribution of a doctoral candidate.

2 Related Work

After a long era of easily predictable fixed voice telephone services, information and communication industry has come to a period of intensive introduction of a very wide spectrum of numerous new services [8]. Rapid technological development and liberalisation have made the information and communication market a very dynamic environment where forecasting is becoming increasingly important. By understanding data patterns during information and communication services' life-cycles, a service provider can perform optimal business planning of its capacities, investments, resources (e.g. human potentials and equipment), marketing and sales. However, there is always the problem of bridging the gap between collected historical data and the anticipated value in the future due to the lack of reliable input data for forecasting.

2.1 Service Growth Models

Every service's life-cycle (SLC) consists of phases shown in Fig. 1 [9]: *development*, *introduction*, *growth*, *maturity* and *decline*. A typical service during its life-cycle passes through specific phases of market adoption, which can be observed through the number of service users. Understanding these phases in a SLC is especially important for highly competitive market environments and particularly for services based on emerging technologies. Numerous researches have resulted in a conclusion that these phases can be described by mathematical models which will be briefly explained here.

Growth models mathematically describe patterns of growth in nature and economy, illustrate how a certain environment reflects on the growth, as well as enable future growth forecasting. Particularly, diffusion of new ideas and technology, market adoption of new products and services, as well as allocations of restricted resources has characteristic S-shaped (sigmoidal) growth. Two most commonly used S-shaped models for initial phases of a SLC (i.e. development, introduction, growth) are the *Logistic* and the *Bass* models. Later phases of a SLC require more complex models (e.g. Bi-Logistic growth model) [8]. This paper will be focused on the services in their initial SLC phases.

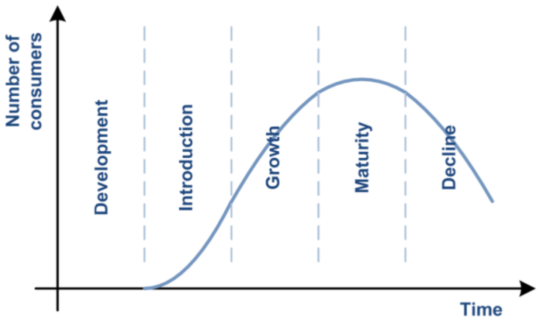


Fig. 1. Information and communication service life-cycle

Logistic Model

The logistic model $L(t)$ is best used for describing growth of the number of service consumers in time in a closed market, isolated from other services. The model is defined with three parameters: M – market capacity, a – growth rate, and b – time shift parameter, as is shown in (1) [10].

$$L(t; M, a, b) = \frac{M}{1 + e^{-a(t-b)}} \tag{1}$$

The logistic model is a widely used growth model with numerous useful properties for technological and market development forecasting. During the first phase, growth of the logistic model is exponential, but later negative feedback slows the gradient of growth as the number of consumers approaches the market capacity limit M. Individual impact of each parameter can be seen in Fig. 2.

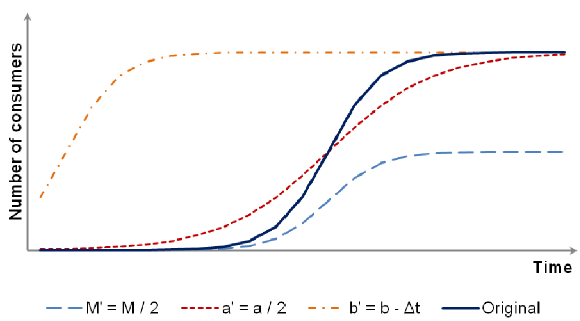


Fig. 2. Interest in a service described by the *Logistic model* with different parameters

Bass Model

The most common model for describing new service diffusion is the Bass model [11]. The Bass model $B(t)$ corrected the deficiency of simple logistic growth (slow growth and no point where $L(t)$ equals zero) by taking into account the effect of innovators via coefficient of innovation *p*. The model divides a population of *M* adopters in two categories: *innovators* (with a constant propensity to purchase) and *imitators* (whose propensity to purchase is influenced by the amount of previous purchasing). Bass diffusion model is defined by the following four parameters:

- M – market capacity;
- p – coefficient of innovation, $p > 0$;
- q – coefficient of imitation, $q \geq 0$, and
- t_s – the moment of service introduction, $B(t_s) = 0$.

These parameters define the model as shown in (2). The Bass model has a shape of S-curve, as does the Logistic model, but the curve is shifted down on the *y*-axis. Fig. 3 shows the effects of different values of parameters p and q on form of S-curve, with fixed values for M and t_s [8].

$$B(t; M, p, q, t_s) = M \frac{1 - e^{-(p+q)(t-t_s)}}{1 + \frac{q}{p} e^{-(p+q)(t-t_s)}} \quad (2)$$

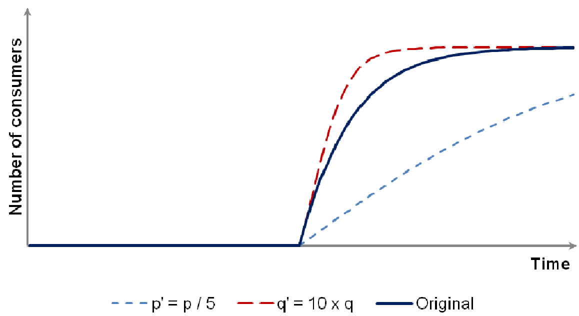


Fig. 3. Interest in a service described by the *Bass model* with different parameters

The Bass model is widely used for the long-term forecasting of a new service market adoption when interaction with other services can be neglected. Furthermore, the Bass model can be used when limited or no data is available (i.e. market adoption forecasting prior to service launch), which is also a focus of our paper. However, while best practices in such cases recommend obtaining model parameters via subjective judgmental assumptions and placing them in optimistic-pessimistic intervals we propose a different, more objective approach. Namely, researches have shown that services with similar characteristics tend to have similar parameters when their growth is represented with the Bass (or Logistic) model [12]. Taking this into account we propose using Semantic Web technologies to profile services and thus enable computers to autonomously calculate the level of similarity between any pair of services.

2.2 Semantic Service Profiling

Semantic markup has proven to be a very efficient method for resource matchmaking and ultimately better recognition of consumer needs. The Semantic Web is a concept in which knowledge is organized into conceptual spaces according to meaning, and keyword-based searches are replaced by semantic query answering. Semantic Web languages, such as Resource Description Framework (RDF), RDF Schema (RDFS) and the Web Ontology Language (OWL), are used to provide structure for resource description (i.e. service profiles) [7].

Using various query languages, based on Structured Query Language (SQL) syntax, it is possible to perform semantic data extraction, thus enabling efficient matchmaking of service profiles once they have been created according to a certain standard. Such matchmaking enables service comparison to be performed according to true, semantic similarities, rather than keyword matchmaking (i.e. YouTube clip tags).

3 Service Modelling System

In order to perform new service growth forecasting it is necessary to ensure mechanisms for four basic processes: i) creating semantic profiles based on service description, ii) mapping existing services' historical data into growth models (i.e. Bass or Logistic model), iii) comparing newly introduced service with existing services, and finally, iv) calculating newly introduced service growth model and corresponding parameters.

Semantic Profiling

First, it is necessary to create an ontology that relates information and communication service descriptions (e.g. tags, multimedia clip reproduction quality and length, etc.) to semantic profiles that can be autonomously processed by computers [7]. It is not possible to create comprehensive service ontology because the number and variety of services on the market increase on an hourly basis. Therefore, it is necessary to ensure a simple ontology upgrade process to enable description of new services, as well matchmaking of new services with those already in the market.

Modelling Existing Services

Number of consumers of a certain services, when observed through time, forms a rather irregular set of discrete data points. Our system must transform this stochastic set of data into a smooth S-curve that approximates the actual numbers with a satisfactory degree of deviation (Fig. 4). Finding the correlation is performed in two steps: recognizing the correct model (e.g. Bass or Logistic model) and calculating the corresponding parameters. For a data model defined by k parameters it is necessary to have at least k known data points. When there is exactly k known data points, the parameters are the solutions of a system of equations. System is usually nonlinear, so iterative numerical methods need to be performed for its solution. When there are more than k data points known, the weighted least squares method is the most commonly used. The objective is to minimise sum of squared differences between actual data points from a real world and evaluated points from a model [8].

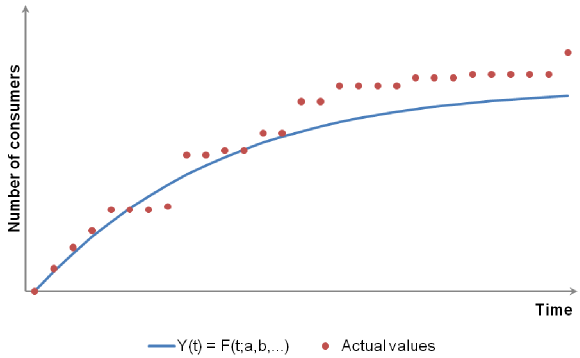


Fig. 4. Transforming actual data points into a model defined as Y(t)

Introducing and Modelling Newly Introduced Services

The final goal of our system is to forecast consumer interest in newly introduced services by calculating its growth model. In order to achieve that it is firstly necessary to see where the newly introduced service fits in the existing set of services on the market. We propose using semantic matchmaking to detect most similar services [7]. Once we have identified similar services it is possible to choose the most appropriate growth model (i.e., the most common growth model among similar services) and calculate the chosen model parameters based on parameter values of the similar services, taking corresponding semantic similarities as weight factors (i.e. the most similar services are the most important while calculating parameter values). This is shown in Fig. 5.

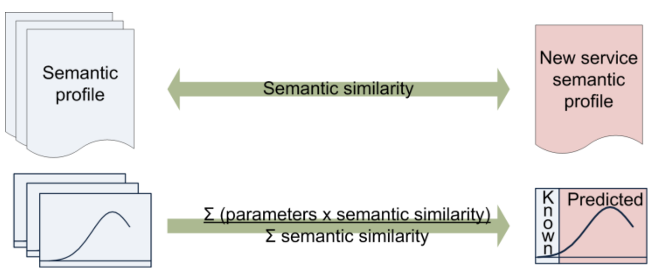


Fig. 5. The process of predicting consumer interest in newly introduced service

4 System Architecture

The system we propose in this paper consists of three main entities: i) *service provider*, ii) *semantic repository*, and iii) *agent-based application* (Fig. 6). The service provider (e.g. YouTube) offers its consumers information and communication services (e.g. multimedia clips) via Internet. Each service is characterised with its description (e.g. identification, category, author, resolution, tags) and data that defines the number of its consumers (e.g. multimedia clip viewers) from the moment it was introduced on the market.

The `Semantic Profiling Agent` transforms service descriptions into semantic profiles. The `Service Modelling Agent` recognizes the adequate growth model for each existing service with sufficient data and calculates the parameters accordingly. The semantic profile and growth model information (for each service) are then stored into a semantic repository so they can later be used for semantic matchmaking with newly introduced services, as well as calculating growth model of newly introduced services. The `New Service Modelling Agent` uses the information stored within the semantic database and calculates the model when a new service is introduced and there is none or insufficient data to calculate the model from its data points.

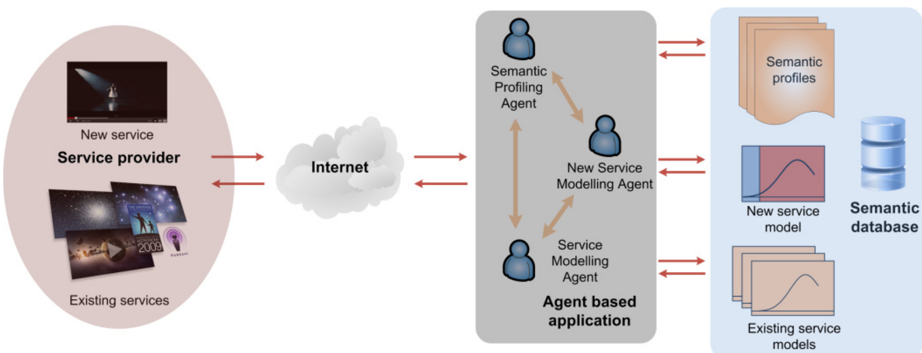


Fig. 6. Architecture of an agent-based system for predicting consumer interest in newly introduced services

5 Proof-of-Concept Scenario: Calculating a Growth Model for a Newly Introduced YouTube Clip

In this section we will present the functionalities of our proposed system. We will use YouTube as a proof-of-concept service provider and multimedia clips as a proof-of-concept information and communication services. YouTube clips are divided into categories (i.e. music, film & animation, education, etc.). Also, each clip has a short description and a set of tags suggested by the author. This information is translated into a semantic profile as described in Section 4. An example of a complete service profile is shown in Fig. 7. The profile consists of four main parts: *identification* (i.e. name, author, URL, etc.), *technical characteristics* (i.e. video clip resolution, frames per second, duration, etc.), *keywords* retrieved from tags (i.e. actor names, director name, content keywords, etc.), and calculated *growth model information* (i.e. model, parameters).

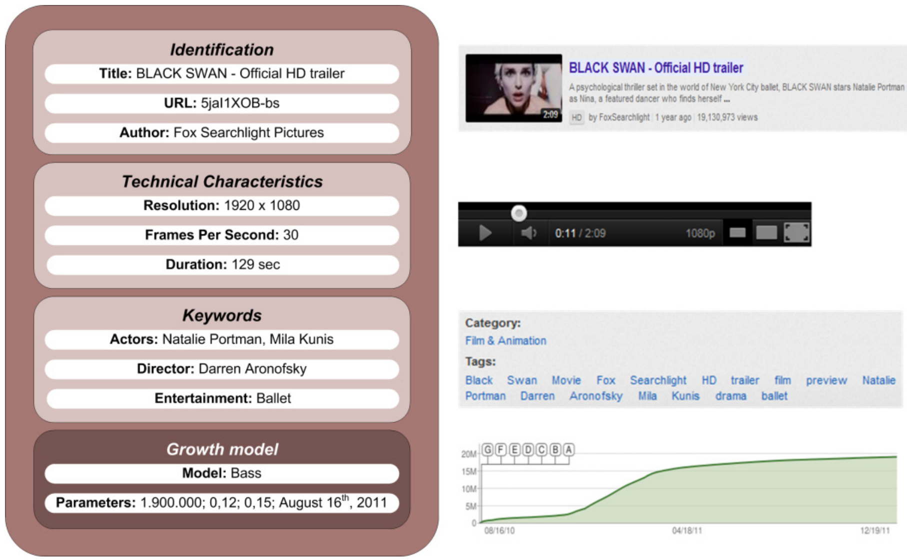


Fig. 7. Semantic profiling of a YouTube clip[1]

Service Matchmaking

Comparison of YouTube clips is based on semantic matchmaking that is performed by the `New Service Modelling Agent` [7]. Each semantic profile consists of attributes and joint values. Semantic matchmaking is based on numeric evaluation of categories clips belong to, basic data type attributes (i.e. resolution, duration), and class instance attributes (i.e. actors, director). Such numeric evaluation results in a final similarity level between 0 and 1.

Semantic matchmaking is first performed between clips' categories. For example, two YouTube clips from *Sports* category are more likely to have similar parameter values than two clips from different categories. After that each common attribute is compared. Numeric value comparison is the quotient of the smaller and the larger number. For example, if we compare clips with durations of 150 and 300 seconds, their similarity is 0.5. String and binary value comparison result is 1 if the values are identical and 0 otherwise. Class instance attributes are compared observing classes' positions in the ontology class hierarchy, where the result is 1 divided by the number of steps between them [7]. For example, if we compare two clips where one has *Ballet* in the *Keyword* section, and the other has *Opera* in the same section, with both *Opera* and *Ballet* representing instances of class *Entertainment*, their similarity will be 0.5.

Once all the attributes have been compared, final profile similarity is calculated using *weighted arithmetic mean* method where weight factors are custom values that represent the importance of each attribute. For example, clip category should have a much higher weight than *Frames per Section* attribute. More detailed insight in semantic matchmaking is presented in [7] and [13].

[1] The information was retrieved from: `http://www.youtube.com/watch?v=5jaI1XOB-bs` (accessed: December 19th, 2011)

Growth Forecasting for Newly Introduced Services

If we were to look back at the time when the *Black Swan Trailer* had just been uploaded, a whole different modelling process would have occurred. When a new video clip is introduced it is compared to all existing clips in the system by the `New Service Modelling Agent`. After semantic matchmaking results are obtained we choose a certain number (e.g. five) of clips most similar to the newly introduced clip. An example of calculated similarities and Bass model parameters for most similar clips is shown in Table 1.

Table 1. Sample clip data

	Semantic similarity	M	p	q
YouTube Clip 1	0.754	2 050 200	0.125	0.132
YouTube Clip 2	0.726	2 433 150	0.139	0.078
YouTube Clip 3	0.719	1 510 220	0.105	0.112
YouTube Clip 4	0.698	2 601 500	0.098	0.102
YouTube Clip 5	0.695	1 896 000	0.117	0.087
YouTube Clip 6	0.631	2 100 500	0.058	0.051
YouTube Clip 7	0.624	4 300 400	0.133	0.035
YouTube Clip 8	0.523	1 500 000	0.081	0.125
YouTube Clip 9	0.496	500 000	0.048	0.193
YouTube Clip 10	0.478	215 000	0.052	0.032
...	...	...	...	...

Parameters for the newly introduced clip are calculated as follows (according to the rule from the Fig. 5):

$$M = \frac{\sum_{i=1}^{5}(s_i \times M_i)}{\sum_{i=1}^{5} s_i} = 2\ 096\ 808 \tag{3}$$

$$p = \frac{\sum_{i=1}^{5}(s_i \times p_i)}{\sum_{i=1}^{5} s_i} = 0.117 \tag{4}$$

$$q = \frac{\sum_{i=1}^{5}(s_i \times q_i)}{\sum_{i=1}^{5} s_i} = 0.103 \tag{5}$$

where s_i represents the semantic similarity between the new clip and YouTube clip i, while M_i, p_i and q_i are Bass model parameters of clip i. Once parameters M, p and q are calculated it is possible to approximate the number of viewers the new clip should reach in near future. As was mentioned earlier, basic Bass model works best if used on initial growth modelling. Latter stages of SLC require more complex models.

6 Conclusion and Future Work

In this paper, we propose a multi-agent system for forecasting consumer interest in new services. The innovativeness of our proposal can be recognized in using semantic reasoning for enhancing newly introduced service growth modelling. The semantic reasoning is particularly helpful when insufficient data about new service popularity is available – semantic reasoning enables us to substitute missing data for parameter calculation with the data from similar services already on the market. Such approach should enable service provider to perform pre-market forecasting in order to determine whether the service has its place in the market or it is destined for failure.

Our future research will be focused on the following three challenges. The first challenge is improving the implemented semantic reasoning mechanism. Key tasks which correspond to this challenge are improving scalability of semantic matchmaking algorithm and creating mechanisms for automated service profiling (e.g. automated transformation of YouTube clip tags into semantic profiles). The second challenge is implementing a more generalized model that will be applicable over complete service life-cycle (not just for the initial life-cycle phases). The final challenge is verification of our proposed system on various information and communication services (e.g. forecasting consumer interest in news based on news diffusion through a social network).

References

1. Vrdoljak, L., Bojic, I., Podobnik, V., Jezic, G., Kusek, M.: Group-oriented Services: A Shift toward Consumer-Managed Relationship in Telecom Industry. Transactions on Computational Collective Intelligence 2, 70–89 (2010)
2. Sreedhar, D., Manthan, J., Ajay, P., Virendra, S.L., Udupa, N.: Customer Relationship Management and Customer Managed Relationship - Need of the hour (April 2011), `http://www.pharmainfo.net/`
3. Schmitt, B.: Customer experience management: a revolutionary approach to connecting with your customers. John Wiley and Sons, Hoboken (2003)
4. Girish, P.B.: How Banks Use Customer Data to See the Future (April 2011), `http://www.customerthink.com/`
5. Shin, N.: Strategies for Generating E-Business Returns on Investment. Idea Group Inc. (2005)
6. Rygielski, C., Wang, J.C., Yen, D.C.: Data mining techniques for customer relationship management. Technology in Society 24, 483–502 (2002)
7. Vrdoljak, L.: Agent System based on Semantic Reasoning for Creating Social Networks of Telecommunication Service Users. Diploma thesis, University of Zagreb, Faculty of Electrical Engineering and Computing, Zagreb, Croatia (2009)
8. Sokele, M.: Analytical Method for Forecasting of Telecommunications Service Life-Cycle Quantitative Factors. Doctoral Thesis. University of Zagreb, Faculty of Electrical Engineering and Computing, Zagreb, Croatia (2009)
9. Sokele, M., Hudek, V.: Extensions of logistic growth model for the forecasting of product life cycle segments. In: Moutinho, L. (ed.) Advances in Doctoral Research in Management, vol. 1, pp. 77–106. World Scientific Publishing (2006)
10. Sokele, M.: Growth models for the forecasting of new product market adoption. Telektronikk 3, 4, 144–154 (2008)
11. Bass, F.: A new product growth for model consumer durables. Management Science 15(5), 215–227 (1969)
12. Tellabs. Forecasting the Take-up of Mobile Broadband Services. White Paper (2010)
13. Li, Y., McLean, D., Bandar, Z., O'Shea, J., Crockett, K.: Sentence Similarity Based on Semantic Nets and Corpus Statistics. IEEE Transactions on Knowledge and Data Engineering 18(8), 1138–1150 (2006)

Comparing Different Overlay Topologies and Metrics in Pulse-Coupled Multi-Agent Systems

Iva Bojic and Mario Kusek

University of Zagreb, Faculty of Electrical Engineering and Computing,
Unska 3, HR-10000, Zagreb, Croatia
{iva.bojic,mario.kusek}@fer.hr

Abstract. In Multi-Agent Systems (MASs) it is of vital importance that basic network operations are performed without the interference of a central entity (i.e. agent). In this paper we will present how to use a self-organization approach to achieve time synchronization of agents in MASs using a Pulse-Coupled Oscillators (PCO) model that is based on flashing fireflies. Fireflies are known to emit flashes at regular intervals when isolated, but when they are within a group, they converge upon the same rhythm until time synchronization is achieved. This paper investigates how the choice of overlay network topology and metric affects the time synchronization process of agents in MASs.

Keywords: self-organization, firefly synchronization, pulse-coupled oscillators, MASON.

1 Introduction

In distributed systems, such as Multi-Agent Systems (MASs), time synchronization is a prerequisite for many processes (e.g. maintaining the consistency of distributed data or checking the authenticity of a request sent to the server). In this paper we present a self-organization approach for time synchronization of agents in MASs using a Pulse-Coupled Oscillators (PCO) model that is based on flashing fireflies. Self-organization is a process in which patterns at the global level of a system emerge solely from numerous interactions among lower-level components in the system [6]. This type of synchronization is applicable in MASs since agents can self-organize themselves and no external control is needed.

The focus in this paper is on investigation how different overlay network topologies and metrics affect the time synchronization process of agents in MASs (i.e. which parameter has a greater impact on the time synchronization process). The choice of an overlay network topology determines the way agents are coupled (i.e. are connected), while the usage of a metric denotes the intensity of influence that coupled agents have on each other.

The rest of this paper is organized as follows. Section 2 introduces the PCO model for time synchronization explaining the meaning of coupling constants and functions. Section 3 gives a short overview of related work and presents our previous work. Furthermore, Section 4 explains the simulation parameters,

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 464–473, 2012.

the evaluation criteria and presents the results. Finally, Section 5 concludes the paper and outlines directions for future work.

2 The Pulse-Coupled Oscillators Model

Time synchronization of agents in MASs can be achieved using the PCO model and can be described with the set $PCO_{MAS} = \{\alpha,\ \phi,\ \Delta,\ \tau,\ \mu\}$. $\alpha = \{a_\imath \mid \imath \in \mathbb{N}\}$ denotes the set of agents in the MAS. The set $\phi = \{\varphi_\imath \mid \imath \in \mathbb{N};\ \varphi_\imath \in [0,\ 1]\}$ contains agents' internal phases. Furthermore, the set $\Delta = \{(x_\imath,\ y_\imath) \mid \imath \in \mathbb{N};\ x_\imath,\ y_\imath \in \mathbb{R}^+\}$ contains all agents' coordinates. Finally, the set $\tau = \{$*fully-meshed, line, meshed, ring, star*$\}$ contains different topologies, while the set $\mu = \{Chebyshev,\ Euclidean,\ Mahalanobis,\ Manhattan\}$ contains different metrics that are used in order to calculate distances among agents. In MASs the dynamic behavior of agent's $a_\imath$ oscillator is given by the following equation

$$z_\imath(t) = f(\varphi_\imath) + \sum_{\jmath=1}^{N} \varepsilon_{\imath\jmath}\ g_{\imath\jmath}(t), \tag{1}$$

where $z_\imath(t)$ is a value of a state variable $z_\imath$ at the moment t, $\varphi_\imath$ denotes agent's $a_\imath$ internal phase, $f(\varphi_\imath)$ describes the *excitation* evolution of agent's $a_\imath$ oscillator, N denotes the number of agents in the MAS, $\varepsilon_{\imath\jmath}$ is a coupling constant, while $g_{\imath\jmath}(t)$ is a coupling function between agents' $a_\imath$ and $a_\jmath$ oscillators.

Theorem 1. ***The Mirollo and Strogatz theorem (1990)*** — *If the function $f(\varphi_\imath) : [0,\ 1] \rightarrow\ [0,\ 1]$ is smooth, monotonically increasing and concave down, then the set of fully-meshed connected agents' oscillators will always converge to synchronicity for any arbitrary set of agents and initial conditions [11].*

The coupling constant $\varepsilon_{\imath\jmath}$ denotes the intensity of the influence that coupled agents $a_\imath$ and $a_\jmath$ have on each other. Mirollo and Strogatz assumed that the coupling constant $\varepsilon_{\imath\jmath}$ is equal for all agents (i.e. $\varepsilon_{\imath\jmath} = \varepsilon,\ \forall\imath\ \forall\jmath$) and proved that synchronization can be achieved if $\varepsilon \in \langle 0,\ 1\rangle$ [11]. In 2010 An et al. proved that synchronization can be achieved even in the conditions of different coupling constants that are distributed in a close interval [1].

Theorem 2. ***The An et al. theorem (2010)*** — *Given two oscillators $a_\imath$ and $a_\jmath$ with their coupling strengths satisfying $\varepsilon_{\imath\jmath} \neq \varepsilon_{\jmath\imath}$, they will achieve synchronization [1].*

The coupling function $g_{\imath\jmath}(t)$ denotes whether agents $a_\imath$ and $a_\jmath$ are coupled and is calculated as

$$g_{\imath\jmath}(t) = \begin{cases} 1, & \textit{if } a_\imath \textit{ is coupled with } a_\jmath \textit{ and } t = t_\jmath^* \\ 0, & \textit{otherwise} \end{cases} \tag{2}$$

where $t_\jmath^*$ is the firing time of agent's $a_\jmath$ oscillator. Mirollo and Strogatz assumed that every agent is coupled with all other agents in the system (i.e. fully-meshed connectivity) [11]. Almost 15 years after their seminal work, Lucarelli and Wang showed that the fully-meshed assumption can be replaced with the partly-meshed assumption and that agents still can achieve time synchronization [9].

2.1 Uncoupled Oscillator

When agent's a_i oscillator is not coupled, its state variable z_i changes following its own *excitation* described with the function $f(\varphi_i)$. Fig. 1 shows an example of the temporal evolution of agent's a_i oscillator *excitation*. When the state variable z_i reaches the *threshold*, agent's a_i oscillator "fires" and then its state variable z_i jumps back to 0 (i.e. $z_i \in [0, \textit{threshold}]$). In the example shown in Fig. 1, the function $f(\varphi_i)$ is a line and the value of *threshold* is set to 1.

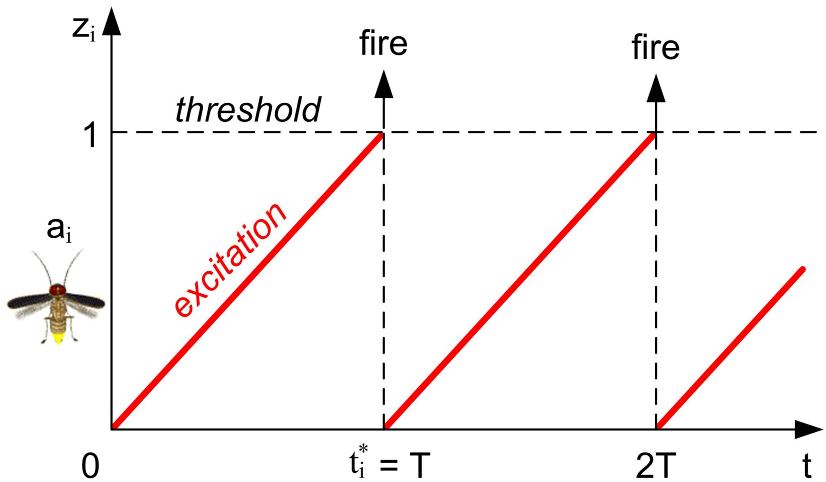


Fig. 1. An example of the temporal evolution of agent's a_i oscillator *excitation*

2.2 Two Coupled Oscillators

The PCO model is based on the experiments conducted on real fireflies by Buck et al. in 1981 [5]. Agent's a_i state variable z_i changes as follows

$$z_i(t_j^*) = \begin{cases} \textit{threshold}, & \textit{if the condition } c \textit{ is met} \\ f(\varphi_i), & \textit{otherwise} \end{cases} . \quad (3)$$

The condition c is met if

$$z_i(t_j^{*-}) \ < \ \textit{threshold} - \textit{buffer} \quad (4)$$

and

$$z_i(t_j^{*-}) + f(\varphi_i) + \sum_{j=1}^{N} \varepsilon_{ij} \, g_{ij}(t_j^*) \ > \ \textit{trigger} \quad (5)$$

where t_j^{*-} denotes the moment before the firing time t_j^* of agent's a_j oscillator.

Fig. 2 shows that at the moment t_j^* condition c is met for agent a_i. Thus, agent a_i sets its state variable z_i to the *threshold* and then flashes. The condition c is met if its state variable z_i at the moment t_j^{*-} is smaller than the value of the expression $(threshold - buffer)$ (Equation (4)) and if the amount of luminescence from the agents that are coupled with agent a_i plus agent's a_i *excitation* is larger than a *trigger* (Equation (5)). Otherwise, agent a_i will not be influenced by the agents that it is coupled with (see Fig. 2 at the moment t_i^*).

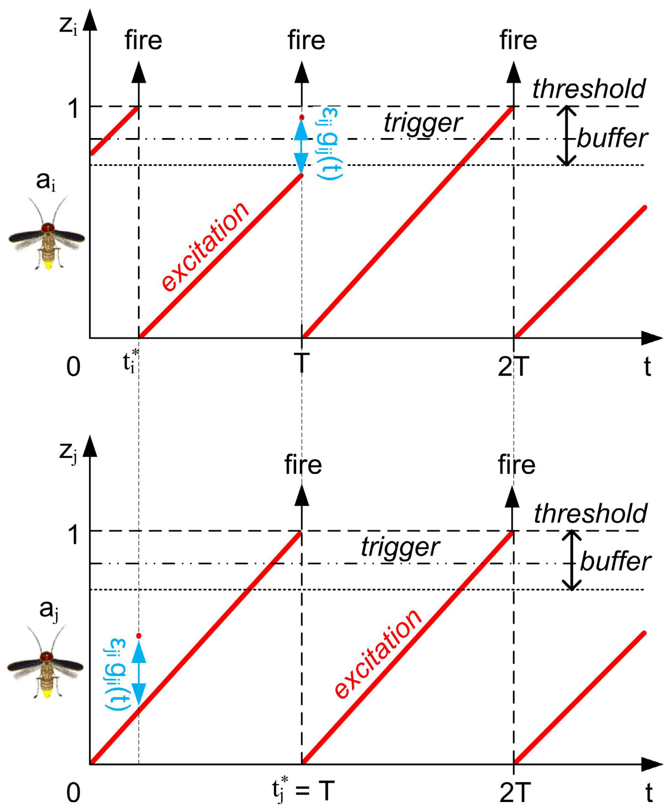


Fig. 2. Example of the temporal evolution of agents a_i and a_j oscillators

The Coupling Function $g_{ij}(t)$. In order to calculate the coupling function $g_{ij}(t)$ it is necessary to find whether agent a_j is coupled with agent a_i at the moment t using information from an overlay network topology. The overlay network topology is defined by agents' coordinates from the set Δ, a type of the overlay network topology from the set τ and a metric from the set μ. In this paper we use fully-meshed, meshed, ring, line and star overlay network topologies (for more details about overlay network topologies see [3]).

The Coupling Constant ε_{ij}. The coupling constant ε_{ij} is inversely proportional to the distance between agents a_i and a_j in the chosen metric. In our previous work [3] [4] we calculated the coupling constant ε_{ij} using only *Euclidean* distance. In this paper we extended the set μ with three more elements: *Chebyshev*, *Mahalanobis* and *Manhattan* distances.

Chebyshev distance

Chebyshev distance is defined on a vector space where the distance between agents a_i and a_j is the greatest of all of their differences along any coordinate dimension. The mathematical expression for *Chebyshev* distance is

$$d_c(a_i, a_j) = \max(\mid x_i - x_j \mid, \mid y_i - y_j \mid). \tag{6}$$

Euclidean distance

Euclidean distance between agents a_i and a_j is the length of the line segment connecting them. *Euclidean* distance can be calculated as

$$d_e(a_i, a_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \tag{7}$$

Mahalanobis distance

When agents a_i and a_j are independent, then *Mahalanobis* distance becomes normalized *Euclidean* distance and can be calculated as

$$d_h(a_i, a_j) = \sqrt{\frac{(x_i - x_j)^2}{var(X)} + \frac{(y_i - y_j)^2}{var(Y)}}. \tag{8}$$

where $var(X)$ is the variance of all agents' x_i, $\forall i$ coordinates calculated as

$$\begin{aligned} var(X) &= \frac{1}{N-1} \textstyle\sum_{i=1}^{N} (x_i - \xi_x)^2 \\ \xi_x &= \frac{1}{N} \textstyle\sum_{j=1}^{N} x_j \end{aligned} \tag{9}$$

and $var(Y)$ is the variance of all agents' y_i, $\forall i$ coordinates calculated as

$$\begin{aligned} var(Y) &= \frac{1}{N-1} \textstyle\sum_{i=1}^{N} (y_i - \xi_y)^2 \\ \xi_y &= \frac{1}{N} \textstyle\sum_{j=1}^{N} y_j . \end{aligned} \tag{10}$$

Manhattan distance

Manhattan distance is based on calculating the distance between agents a_i and a_j as a sum of absolute differences of their coordinates. The mathematical expression for *Manhattan* distance is

$$d_n(a_i, a_j) = \mid x_i - x_j \mid + \mid y_i - y_j \mid . \tag{11}$$

3 Related Work

The Mirollo and Strogatz model [11] gave a theoretical framework for the convergence to synchrony in fully-meshed networks, while Lucarelli and Wang showed that the algorithms developed by Mirollo and Strogatz would converge for any connected topology [9]. Hence, the Mirollo and Strogatz model considers that there exists physical connectivity among all agents in the system, while the Lucarelli and Wang model specifies the interconnection in the network with a graph in which edges join connected agents.

In our previous work [3] [4] we investigated what is the effect that different overlay network topologies (i.e. fully-meshed, meshed, ring, line and star) have on the time synchronization process. This paper extends our previous work by adding different metrics denoting a *credibility* measure. These two parameters have already been studied in related work, but to our knowledge, no previous study compared their mutually effect on the time synchronization process.

3.1 Overlay Network Topologies

The usage of the PCO model for time synchronization can be found in various algorithms (e.g. geographic routing algorithms [12] and the Reachback Firefly Algorithm (RFA) [15]), protocols (e.g. gossip protocols [16]) or data gathering mechanisms (e.g. data gathering scheme in sensor networks [14]). All of the aforementioned examples assumed meshed networks in which all agents were the same and had the same influence on each other (i.e. $\varepsilon_{\imath\jmath} = \varepsilon,\ \forall\imath\ \forall\jmath$).

In every step of a geographic routing [12], an agent can communicate only with its k-neighbors, referred to as the Connected k-Neighborhood (CKN) problem. This k-connectivity is also used in [2], where for every agent, k-neighbors are selected randomly. In the RFA four different topologies have been investigated: fully-meshed and meshed [15], chain [8] and ring [7]. In the chain topology, agents are ordered in a chain and can communicate only with their immediate neighbors, while the ring topology was simulated using asynchronous communication patterns (i.e. unidirectional communication links).

3.2 *Credibility* Measures

Although, in most related work the equal *credibility* is assumed for all agents, there are projects where the coupling constant $\varepsilon_{\imath\jmath}$ is different for different agents. The usage of the *credibility* measure enables different agents to have different influence on others. When agents (e.g. attackers) are less credible, then they have less influence on other agents. Therefore, the *credibility* measure can be used as a means to increase the robustness of the time synchronization process.

In [13] authors proposed that agent's *credibility* depended on the degree-based weighting. Being influenced by more agents means that the agent is likely to be more reliable. Furthermore, An et al. assumed that all agents' coupling constants stayed constant during the time synchronization process and were distributed within a close interval. They proved that under those conditions time synchronization can be achieved [1].

4 The Simulation of Time Synchronization in the MAS

Our simulation is run using a single-process discrete-event simulation core and visualization toolkit written in Java called Multi-Agent Simulator of Neighborhoods (MASON). In this paper, we will not explain how agents' activity is scheduled in the simulator, nor other details about the simulation process, since it can be find in the MASON's documentation [10].

4.1 Simulation Rules

The behavior of each agent in simulation is composed of two parts: *active* and *passive* one. In the active part, the value of state variable $z_\imath$ increases until it reaches the *threshold* (i.e. $z_\imath = f(\varphi_\imath)$), at which point a flash is emitted and then $z_\imath = 0$. Since the flashes are timed by the progressive excitation (described with the function $f(\varphi_\imath)$) within each agent, when left alone, each agent flashes at periodic intervals. In the passive part, each agent senses a certain amount of luminescence from the agents that it is coupled with and reacts upon it according Equations (3), (4) and (5).

4.2 Simulation Parameters

The simulation parameters include *numberFireflies*, *threshold* and *buffer*. The *numberFireflies* parameter sets the number of fireflies in the simulation. Furthermore, the *threshold* parameter denotes at which point an agent will flash and reset its state variable to zero. Finally, the *buffer* parameter sets how many time steps are necessary for the flashing signal to evolve and terminate in a flash. If an agent is triggered to reset its state variable while condition from Equation (4) is not met, the flash will proceed as planned despite the resetting.

In this paper we simulate small overlay networks with ten agents, and therefore the *numberFireflies* parameter is set to 10. In order to determine values of *threshold* and *buffer* parameters, in our previous work [3] we made simulation experiments. We changed the ratio between *threshold* and *buffer* parameters from 0.1 to 0.9. The conclusion was that the best ratio is 0.1. This means that the relationship between *threshold* and *buffer* parameters can be described with the following equation: *buffer* = 0.1 *threshold*. Therefore, in this paper we choose that the *threshold* value is equal to 20 and the *buffer* value is equal to 2.

4.3 Evaluation Criteria

The evaluation criteria are the same as in our previous papers [3] [4]: the percentage of time synchronization success, the network traffic and the time needed to achieve time synchronization. Agents are said to be synchronized when they flash simultaneously. The network traffic denotes the amount of messages exchanged among agents during the time synchronization process, while the time needed to achieve time synchronization denotes discrete time units (i.e. steps) until all agents are synchronized.

4.4 Simulation Results

In our previous work [3] we used only *Euclidean* distance with combination of five different overlay network topologies and concluded that best topologies on condition of the percentage of time synchronization success are fully-meshed, meshed and star topologies. In this work we extend our previous results using new metrics. For the purpose of testing the influence of topologies and metrics on the time synchronization process we used combinations of all metrics and topologies (i.e. forty combinations). Each combination is executed 100 times and then average results are shown in Tables 1 and 2.

First we investigate how different metrics affect the percentage of time synchronization success. Results in Table 1 show that *Manhattan* distance is resulting with the best result. For *mesh(3)* topology, when using *Manhattan* distance, results are 10% better than when using *Chebyshev* distance.

Furthermore, we investigate which parameter is more important: the overlay network topology or the metric. From Table 1 it can be concluded that the choice of overlay network topology has a greater influence on the percentage of time synchronization success than the chosen metric. The most significant difference between different overlay network topologies is between fully-meshed and ring topologies when using *Euclidean* distance (27%). The most significant difference between different metrics is between *Chebyshev* and *Manhattan* distances for *mesh(4)* topology (12%).

Finally, we measure the time needed to achieve time synchronization and the number of exchanged messages considering the usage of different combinations of overlay network topologies and metrics. Form the results shown in Table 2, we can conclude that the choice of metric does not have a significant impact on those two parameters.

Table 1. The percentage of time synchronization success

		Metrics			
		Chebyshev	Euclidean	Mahalanobis	Manhattan
Topologies	Fully-meshed	94%	97%	94%	96%
	Line	70%	69%	72%	72%
	Mesh(3)	74%	78%	78%	84%
	Mesh(4)	77%	84%	84%	89%
	Mesh(5)	91%	93%	93%	93%
	Mesh(6)	89%	92%	92%	93%
	Mesh(7)	88%	94%	94%	96%
	Mesh(8)	96%	93%	93%	95%
	Ring	71%	70%	71%	72%
	Star	94%	94%	94%	94%

Table 2. Average time needed to achieve time synchronization / average network traffic during the time synchronization process

		Metrics			
		Chebyshev	Euclidean	Mahalanobis	Manhattan
Topologies	Fully-meshed	47/217	42/198	47/217	43/202
	Line	128/83	128/83	128/83	128/83
	Mesh(3)	64/76	64/77	64/77	68/83
	Mesh(4)	54/92	56/97	56/97	56/98
	Mesh(5)	54/124	54/123	54/123	53/120
	Mesh(6)	49/139	48/136	48/136	46/130
	Mesh(7)	47/159	46/157	46/157	46/158
	Mesh(8)	49/197	45/183	45/183	44/175
	Ring	95/70	95/70	95/70	95/70
	Star	40/29	40/29	40/29	40/29

5 Conclusions and Future Work

This paper proposes a time synchronization scheme for Multi-Agent Systems (MASs) using fireflies as role models. In nature fireflies are known to emit flashes at regular intervals when isolated, while in a group they achieve synchronization. In computer science, fireflies behavior is modeled with a Pulse-Coupled Oscillators (PCO) model. This model can be used in MASs since every agent can run an algorithm similar to the one run by fireflies in nature. Moreover, the PCO model is said to be robust, scalable and adaptive, all of which are important in MASs.

In this paper we tested our algorithm with different overlay network topologies and metrics. Information from overlay network topologies is used in order to find coupled agents, while metrics are used in order to calculate the intensity of influence that coupled agents have on each other. From results presented in this paper, it can be concluded that time synchronization can be achieved when using different coupling constants and moreover, that the choice of overlay network topology is more important than the choice of used metric.

The goal of my Ph.D. research is to design a firefly algorithm that shows a high percentage of robustness without a significant decrease in the percentage of time synchronization success. In order to achieve my goal I will use the coupling constant ε_{ij} as a parameter that denotes agents' *credibility* such that some agents (e.g. attackers) are less credible and thus have less influence on other agents. Therefore, the focus in this paper was on investigation of how coupling constants affected the time synchronization process.

Acknowledgements. The authors acknowledge the support of research project "Content Delivery and Mobility of Users and Services in New Generation Networks", funded by the Ministry of Science, Education and Sports of the Republic of Croatia and "Machine-to-Machine Communication challenges", funded by Ericsson Nikola Tesla, Croatia.

References

1. An, Z., et al.: Linear Pulse-Coupled Oscillators Model: A New Approach for Time Synchronization in Wireless Sensor Networks. Advances in Complex Systems 2(2), 108–114 (2010)
2. Babaoglu, O., et al.: Firefly-Inspired Heartbeat Synchronization in Overlay Networks. In: Proceedings of Self-Adaptive and Self-Organizing Systems Conference, pp. 77–86 (2007)
3. Bojic, I., et al.: A Self-Optimizing Mobile Network: Auto-Tuning the Network with Firefly-Synchronized Agents. Information Sciences 182(1), 77–92 (2012)
4. Bojic, I., Kusek, M.: Fireflies Synchronization in Small Overlay Networks. In: Proceedings of Information and Communication Technology, Electronics and Microelectronics Conference, pp. 27–32 (2009)
5. Buck, J., et al.: Control of Flashing in Fireflies. Journal of Comparative Physiology A 144, 287–298 (1981)
6. Camazine, S., et al.: Self-Organization in Biological Systems. Princeton University Press (2003)
7. Leidenfrost, R., Elmenreich, W.: Establishing Wireless Time-Triggered Communication Using a Firefly Clock Synchronization Approach. In: Proceedings of Intelligent Solutions in Embedded Systems Workshop, pp. 1–18 (2008)
8. Leidenfrost, R., Elmenreich, W.: Firefly Clock Synchronization in an 802.15.4 Wireless Network. EURASIP Journal on Embedded Systems (7), 1–17 (2009)
9. Lucarelli, D., Wang, I.J.: Decentralized Synchronization Protocols with Nearest Neighbor Communication. In: Proceedings of Embedded Networked Sensor Systems Conference, pp. 62–68 (2004)
10. MASON web site, http://www.cs.gmu.edu/~eclab/projects/mason/
11. Mirollo, R.E., Strogatz, S.H.: Synchronization of Pulse-Coupled Biological Oscillators. SIAM Journal on Applied Mathematics 50(6), 1645–1662 (1990)
12. Nath, S., Gibbons, P.: Communicating via Fireflies: Geographic Routing on Duty-Cycled Sensors. In: Proceedings of Information Processing in Sensor Networks Conference, pp. 440–449 (2007)
13. Scholtes, I., et al.: Epidemic Self-Synchronization in Complex Networks of Kuramoto Oscillators. Advances in Complex Systems 13(1) (2010)
14. Taniguchi, Y., Wakamiya, N., Murata, M.: A Distributed and Self-Organizing Data Gathering Scheme in Wireless Sensor Networks. In: Proceedings of Information and Telecommunication Technologies Symposium, pp. 299–304 (2005)
15. Werner-Allen, G., et al.: Firefly-Inspired Sensor Network Synchronicity with Realistic Radio Effects. In: Proceedings of Embedded Networked Sensor Systems Conference, pp. 142–153 (2005)
16. Wokoma, I., et al.: A Weakly Coupled Adaptive Gossip Protocol for Application Level Active Networks. In: Proceedings of Policies for Distributed Systems and Networks Workshop, pp. 244–247 (2002)

Network-Centric Operations in Machine-to-Machine Networks

Damjan Katusic and Gordan Jezic

University of Zagreb, Faculty of Electrical Engineering and Computing,
Department of Telecommunications, Zagreb, Croatia
{damjan.katusic,gordan.jezic}@fer.hr

Abstract. This paper explores the convergence of Machine-to-Machine (M2M) networks with Network-Centric Operations (NCO). An overview of the M2M communication paradigm and current standardization efforts regarding architecture are given. It includes an overview of the network-centric networking and analyses the idea of implementing NCO approach to achieve situational awareness and self-synchronization of autonomous and intelligent M2M devices in networked M2M environments. Case study involving M2M e-Health scenario is given and discussed as a proof of concept for the proposed convergence.

Keywords: machine-to-machine, network-centric operations, information sharing, situational awareness, self-synchronization, software agents.

1 Introduction

As stated in [1], M2M (*Machine-to-Machine*) technology, sometimes also referred as MTC (*Machine Type Communication*), is based on the idea that a machine has more value when it is networked and that the network becomes more valuable as more machines are connected. Even today, number of connected M2M devices is measured in hundreds of millions, and according to Harbor Research number is expected to increase to 390 million cellular connections in 2014 [2]. Mobile Market Development has in [3] analysed trends and opportunities in M2M market, and came up with a projected compound annual growth rate of 25% through 2014, and a projected 50 billion M2M devices worldwide by 2025 [3]. Ericsson's numbers have been even more rapid; according to their predictions, number of 50 billion devices will be reached around 2020 [4]. M2M systems will not just include billions of devices, they will offer MNOs (*Mobile Network Operator*) worldwide ability to extend their businesses, provide new services, and bring billions of dollars of new revenue streams. Juniper research predicts that considering the variety of different M2M applications and products, mobile and embedded M2M market will reach nearly $35 billion by 2014 [5].

Both ETSI (*European Telecommunications Standards Institute*) and 3GPP (*Third-generation Partnership Project*) give very similar definitions of M2M communication.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 474–483, 2012.

According to their publications, it is the communication between two or more entities that do not necessarily need any direct human intervention [6] [7]. Actors in such an environment can include broad range of communication capable devices: computers, mobile phones, tablets, but also variety of sensors, smart grid networks, embedded processors, industrial and medical equipment, and countless other devices. Such diversity offers new business concepts in numerous market verticals, offering a truly networked environment with seamless and automated flow of data and services.

This paper is organized as follows. In Section 2, we give a brief overview of standardization efforts in the M2M domain, while Section 3 brings prominent considerations in the area of M2M architectures. Section 4 focuses on the analysis of integrating NCO (*Network-Centric Operations*) into M2M networks, explains main features of the approach and discusses its importance for M2M communication, particularly information sharing and management. In Section 5, we give a brief case study considering e-Health M2M scenario in the context of analysed NCO networking. Finally, Section 6 concludes the paper.

2 M2M Standardization Efforts

Early 1990s witnessed the development of first SCADA (*Supervisory Control And Data Acquisition*) systems [8]. These primitive industrial management and telemetry systems are usually perceived as precursors of modern M2M systems. They include many connected sensors from which they gather data. However, their biggest disadvantage is high cost because they are based on proprietary communication technologies. Unlike SCADA, M2M systems work with standardized technologies [1] and are independent of the used access in a wide range of possible wired or wireless network technologies (DSL (*Digital Subscriber Line*), Wi-Fi, ZigBee, Bluetooth, cellular, satellite etc.). Current wireless networks are optimized for H2H (*Human-to-Human*) interactions, and communication in M2M systems can greatly differ from such traffic patterns. It is important that mentioned technologies evolve and develop competitive capabilities to efficiently support M2M communication, but in a universal and collaborative manner. Many standardization bodies, including mentioned ETSI and 3GPP, have recently actively engaged in M2M standards development: IEEE (*Institute of Electrical and Electronics Engineers*), 3GPP2 (*3rd Generation Partnership Project 2*) and TIA (*Telecommunications Industry Association*) [9].

ETSI produces globally applicable standards for ICT (*Information and Communications Technology*) including fixed, mobile, radio, Internet, and other areas, and continually strives to improve collaboration with other research bodies [10]. In 2007, a new ETSI technical committee for developing standards for M2M communication has been established. The group aims to provide an end-to-end view of M2M standardization, closely co-operates with ETSI's activities on NGN (*Next Generation Networks*), and also with the work of the 3GPP standards initiative for mobile communication technologies. Most of their standards regarding M2M technology analyze different use cases. Anyhow, they have also put some effort into defining basic M2M terminology, as well as its service and functional requirements [6].

3GPP produces highly successful reports and specifications that define 3GPP technologies, from GSM (*Global System for Mobile Communications*), GPRS (*General Packet Radio Service*) and EDGE (*Enhanced Data rates for GSM Evolution*) systems towards UMTS (*Universal Mobile Telecommunications System*), and recently LTE (*Long Term Evolution*) [11]. Most of its M2M standardization efforts conduct within several Service and System Aspects Working Groups (SA WG): SA WG1, SA WG2, and SA WG3. WG1 focuses on services[1], WG2 on architecture[2], and WG3 on problems regarding security[3]. Two additional groups are actively working on the improvements for radio access networks: GERAN (G*SM EDGE Radio Access Network*) WG2 on protocol aspects of GPRS/EDGE networks[4], and RAN (*Radio Access Network*) WG2 on Layer 2 and Layer 3 Radio Resource specification.[5]

3 M2M Network Architecture Considerations

Broad market potential of M2M networks is consequence of its numerous possible applications and use cases, as well as the variety of available access technologies that can be used in its implementation. This networks need to be reliable, scalable, secure, and manageable. The easiest way to accomplish this, and to solve set of unique challenges that differentiate them from H2H networks (large number of connected devices, specific traffic patterns, many types of devices, small energy requirements etc.) is standardization. One of its main aspects is consideration regarding architecture.

3.1 ETSI Architecture Standardization Approach

ETSI TC (*Technical Committee*) M2M [6] published high-level architecture concept for M2M application support (Fig. 1). It includes network (*Access Network*, *Core Network*, *Service Capabilities*, *M2M Applications*, *Network Management*, and *M2M Management Functions*), and device and gateway domains (*M2M Devices*, *M2M Gateway*, and *M2M Area Network*, including *M2M Applications* and *M2M Service Capabilities*). M2M Devices include broad range of devices cited in the introduction, run M2M Application(s) using M2M Service Capabilities, and are capable to autonomously exchange data with other devices. They connect to network domain in two different ways: directly or through M2M Gateway which serves as a network proxy. M2M Gateway uses M2M Capabilities to ensure M2M Devices are

---

[1] 3GPP Specification series 22, `http://www.3gpp.org/ftp/Specs/html-info/22-series.htm`

[2] 3GPP Specification series 23, `http://www.3gpp.org/ftp/Specs/html-info/23-series.htm`

[3] 3GPP Specification series 33, `http://www.3gpp.org/ftp/Specs/html-info/33-series.htm`

[4] 3GPP Specification series 37, `http://www.3gpp.org/ftp/Specs/html-info/37-series.htm`

[5] 3GPP Specification series 43, `http://www.3gpp.org/ftp/Specs/html-info/43-series.htm`

interworking and interconnected to the underlying communication network, and can provide various services to them. M2M Area Network connects gateway and devices that lack service capabilities and are not capable to directly connect to Access Network. Network domain comprises of Access and Core Networks, enables communication between M2M Gateways and M2M Applications, and includes Network and M2M Management Functions. Access Networks include (but are not limited to): DSL, satellite, GERAN, UTRAN (*Universal Terrestrial Radio Access Network*), eUTRAN (*evolved UTRAN*), WLAN (*Wireless Local Area Network*) and WiMAX (*Worldwide Interoperability for Microwave Access*). Core Networks (CN) include (but are not limited to) 3GPP CNs, ETSI TISPAN (*Telecoms & Internet converged Services & Protocols for Advanced Networks*) CN and 3GPP2 CN. M2M Applications run the service logic and use M2M Service Capabilities accessible via an open interface.

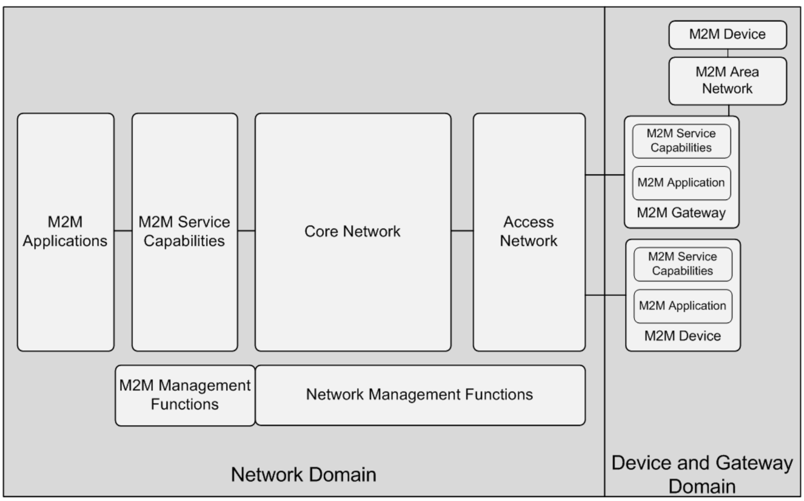


Fig. 1. ETSI M2M Network Architecture

3.2 3GPP Architecture Standardization Approach

Recent developments in wireless industry such as widespread availability of wireless connectivity, declining prices of M2M modules, and regulatory incentives for certain industries (smart grid, e-Health) have attracted attention of potential M2M stakeholders [9]. With the merit of providing higher-layer connections, 3GPP network scenarios have been regarded as one of the most promising M2M solutions [8].

MTC Device comprises of variety of devices mentioned in the introduction and connects via MTC_u interface to the 3GPP network (GERAN, UTRAN, E-UTRAN etc.). The 3GPP network provides transport and communication services (including 3GPP bearer services, IMS (*IP Multimedia Subsystem*), and SMS (*Short Message Service*)) optimized for the MTC communication that connect MTC Device with a

MTC Server or other MTC Devices. The MTC Server connects to the 3GPP network via MTC_i/MTC_{sms} interface and thus communicates with MTC Devices [7] (Fig. 2).

There are three scenarios regarding the communication between MTC Server and MTC Devices [7]. First involves MTC Devices communicating with one MTC Server (n-to-one), which can be located inside or outside operator domain, while in second there are many MTC Servers communicating with many MTC Devices (n-to-n). Third scenario consists of MTC Devices communicating directly with each other without intermediate MTC Server. The latter scenario is not included in 3GPP standardization efforts, because relevant applications only seem to involve device-to-server communication. However, this scenario contains many possibilities that should greatly increase the value and potential of M2M networks in many areas, including information management, which is further discussed in the following chapters.

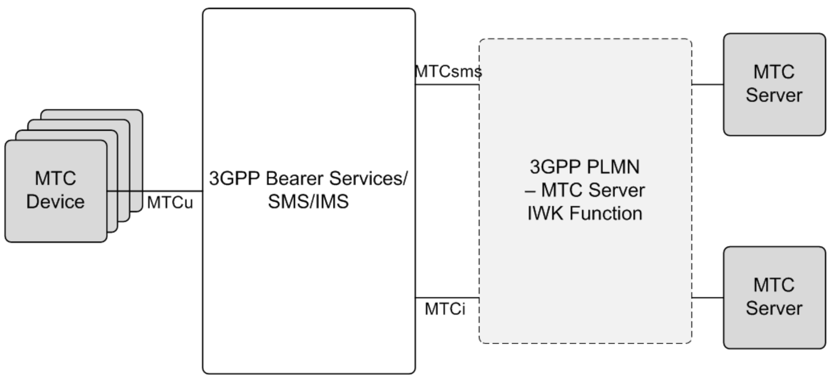


Fig. 2. 3GPP M2M Network Architecture

4 Network-Centric Approach in M2M Networks

M2M networks are due to its diversity still faced with information management and coordination challenges. They can consist of various types and numbers of devices, local or wide area coverage, different level of mobility, autonomy, intelligence or energy constraints. Information management in such an environment can be based on one of the two basic types, centralized or distributed. In centralized management system is continuously examining the environment using its sensor capabilities, transmitting the gathered data to the central node (MTC Server) for processing, and eventually decision making. Commands are distributed from centre to the edges [12]. This approach is correlated to the MTC Device(s)-MTC Server(s) 3GPP scenarios. In distributed information management approach, inherent to the NCO principles, control and intelligence is shifted from one node to the whole network, or as it is in [13] called, “infostructure”. The distribution of control and intelligence among many nodes allows processing of gathered data within the network, which is in relation to the MTC Device-MTC Device 3GPP scenarios. NCO features motivate the idea of integrating some of its concepts in the perspective area of machine type communications.

4.1 Network-Centric Operations

Network-centric operations refer to a continuously-evolving, complex community of people, devices, information, and services interconnected by a communications network in order to optimize resource management and provide superior information on events and conditions needed to empower decision makers [14]. The term "network-centric" is attributed to the U.S. Admiral Jay Johnson who used it to describe what he has seen as a "fundamental shift from platform-centric warfare to network-centric warfare" [13]. Network-centric operations approach is in direct opposition to platform centricity, it shifts from viewing actors as independent to viewing them as part of a continuously adapting ecosystem, and emphasizes the importance of making strategic decisions to adapt or even survive in such a changing environment [13] [15]. Its underlying framework has been influenced by certain command and control processes characterized by an iterative sequential series of steps, such as OODA (*Observe, Orient, Decide and Act*) loop cycle attributed to former USAF (*United States Air Force*) Colonel John Boyd, a model consisting of sense, process, compare, decide, and act steps, developed by Dr. Joel S. Lawson, and the HEAT (*Headquarters Effectiveness Assessment Tool*) process developed by Dr. Richard E. Hayes in 1984 [15].

Network-centric approach seeks to achieve an information advantage, enabled in part by information and communication technologies, into a competitive advantage through the robust networking. This networking, combined with changes in technology, organization, processes, and people, enables organizations to behave and respond in ways never before possible [16]. Specifically, the NCO concept contains the following four tenets, as proposed in [17]:

- A robustly networked force improves information sharing.
- Information sharing enhances the quality of information and shared situational awareness.
- Shared situational awareness enables collaboration and self-synchronization, and enhances sustainability and speed of command.
- These, in turn, dramatically increase mission effectiveness.

These tenets can be, with certain adjustments, applied to a networked environment of any kind. For a success it is critical to achieve qualitative information sharing that eventually enables shared situational awareness and self-synchronization. This, simply speaking, means to know as much relevant aspects of the networked operational environment as possible (e.g. the position of resources, the status of each job).

4.2 Network-Centric Operations in M2M Networks

NCO paradigm described in the previous section mentioned several important features. However, despite the necessary technological basis that allows all of these features to fit together, the most important variable is behaviour of a decision maker and its ability to make use of acquired information and knowledge. Humans, characterized by their cognitive processes and abilities, as well as their social interactions and organization, are the final destination of almost all of the acquired intelligence and knowledge through networking. NCO results are ultimately driven and evaluated by human reasoning. Nodes in a network-centric environment are autonomous decision making units that can collaborate with other nodes (serve other units or be served by them). Devices in M2M environment are capable of same type of behaviour: function autonomously, without the direct human intervention, and are characterized by different cognitive and learning abilities. In such an environment they do not only support the network to process data for human decision makers, but are at some level decision makers themselves. They are driven to reach goals through autonomous interactions with other devices and autonomous decision making.

Information Sharing

Insight into the importance of information sharing is provided by Metcalfe's Law, which describes the potential value of a network. It states that as the number of nodes in a network increases linearly, the potential "value" or "effectiveness" of the network increases exponentially as the square number of nodes [15]. Therefore, more nodes mean more information that can be used to achieve set goals. The NCO paradigm was devised in the first place to allow users to extract qualitative information in the networked environment for better decision making. Situation is the same if M2M devices are the decision makers: network-centricity implies the necessity for better information connectivity between M2M network nodes in such an environment. There are numerous ways to achieve this, and research presented in [18] suggests that adapted publish/subscribe middleware is a potential candidate for the information management solution in networked infrastructures.

Self-synchronization

Self-synchronization is the notion that highly complex groups, with accurate detailed information available at all levels, organize naturally (and optimally) from the bottom up [19]. Self-synchronization of individuals within groups is inherent to most humans or animals (e.g. flocking algorithms etc.), and can also be achieved in M2M systems. Then it allows proactive planning and the ability to make decisions based on information and knowledge gathered through sensors, stored in databases, and analyzed between many intelligent nodes in network at all levels. It creates new operational capabilities through autonomous cooperation of nodes and allows better decision making.

Decision Making

Finally, the last important aspect of NCO, and the desired consequence of all mentioned features, is the smart decision making. Human societies are accustomed to hierarchical organization and highly-centralized top-down commands. NCO changes that notion and pushes shared situational awareness to the edge of the network. This disrupts traditional practices in top-down command and control environments, especially military [20], which is one of the pioneers of NCO research. Firstly, information non-attribution reverses the assumption that commands are issued from an individual entity to a particular individual entity: they are issued to a pool with rather undefined responsibilities. Secondly, as a consequence of such decentralization, decision making is migrating to the edges of a network, giving access to information of quality and quantity that is potentially equal to or better than that available at the centre. US Army's restructuring into smaller units, such as formation of Stryker brigades [21], is evidence that even strict hierarchical human organizations have started to accept benefits of such an approach. This same decentralization can also be applied in M2M environment, nevertheless 3GPP prefers server-client architecture [7], allowing end-devices to enrich it and to achieve full utilization of the available information sharing. One of the main prerequisites for accomplishing this is implementation of a multi-layered and robust information grid within the M2M network that allows sharing of structured data, information, and knowledge.

5 Case study: M2M E-Health Scenario

Healthcare M2M systems will be according to projections in [2] one of the main market drivers of the global M2M growth in the following years. Healthcare is an information rich and knowledge intensive environment, and in order to treat and diagnose even a simple condition, a physician must combine many varied data elements and information [22].

This analysis presumes that the architectural framework of the observed M2M e-Health scenario is largely based on the ETSI Remote Patient Monitoring model drafted in [22], with the inclusion of 3GPP cellular network as its underlying network access and core solution. Such a system includes remote patient monitoring M2M Devices connected via Home Hub which acts as a M2M Gateway to the clinical side. Patient device gathers patient measurements, data and/or events. Data may be communicated each time device gathers data, accumulated measurements may be communicated periodically (e.g., hourly, daily), or data may be delivered upon request or upon certain events. Clinical side involves care coordinator M2M services that monitor received patient data (M2M Server), and if measurements indicate that there has been a change in the patient's health status, or fall outside of a predetermined range, they alert clinician personnel (physician, nurse etc.). This study for the more complete e-Health picture also involves some other healthcare stakeholders that are not crucial for the remote monitoring case, such as pharmacy and healthcare insurance (Fig. 3).

Presently, many existing e-Health initiatives and M2M networks in general represent distinctive and loosely connected entities whose operation is still largely platform-centric, that is, concentrates on the operations of a single system (platform) with too little regard for the operational interaction among different systems [22].

Mentioned platform-centricity has major influence on the access range and sharing of information stored among (or even within) the existing individual systems, and calls for another approach: implementing a network-centric system that facilitates information sharing among all participants within the operational space. Such an approach proposes the creation of multilayered exchange grid that will allow free flow of information, in observed case among healthcare actors. This approach pushes critical information and shared situational awareness to nodes that need it to successfully accomplish tasks, no matter where in network are they.

Satisfying "right information to the right place at the right time" concept, while taking into account QoS (*Quality of Service*) requirements, is not a trivial task. Object Management Group's DDS (*Data Distribution Service*) offers few important ideas. This QoS-enabled data-centric middleware platform enables communication through

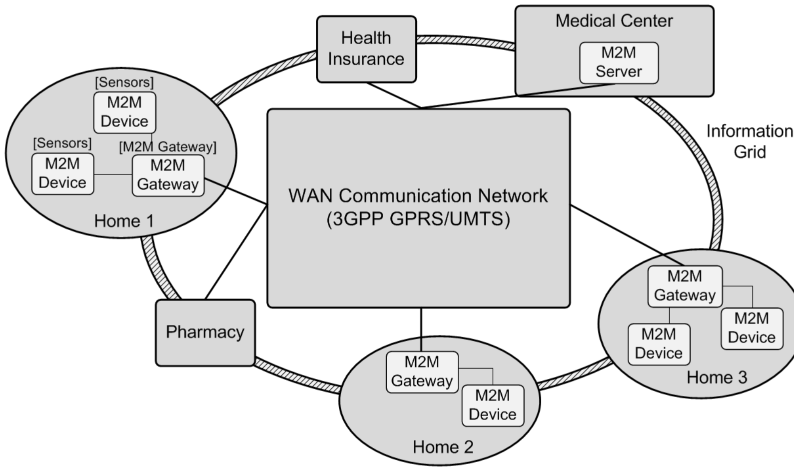


Fig. 3. E-Health M2M-NCO Scenario

real-time publish/subscribe model. It supports location independence (via anonymous pub/sub protocols), scalability (through support of large number of data readers/writers and topics), and platform-portability (via standardized interfaces) [24].

M2M systems on top of 3GPP mobile networks are heterogeneous environments that could greatly benefit from the establishment of such a network-centric, dynamic, and collaborative information management system. There are few important capabilities that such system should support: universal access to information (from a variety of sources), orchestrated information bus (that connects all M2M devices and aggregates, filters, and prioritizes information delivery), continuous adaption to changes (to dynamic network topologies, M2M network membership changes etc.), and support defined QoS policies and mechanisms [24].

Mobile agent and multi-agent software paradigms offer a possible approach for information management framework implementation and enrichment of its characteristics. They employ features such as autonomy, intelligence, cognitive reasoning, learning, proactive actions, and communication ability that have proved over the years in various solutions in distributed and networked environments.

6 Conclusion

M2M networks are due to its diversity still faced with management and coordination challenges. NCO as an information management concept is a way of maximizing the value of M2M solutions, moving from centralized to decentralized control and reaching the maximum utilization of available information. Implementing such an approach proposes the development of interconnected, robust and dynamic information grid within the M2M network infrastructure that will increase shared situational awareness and allow self-synchronization of connected smart devices. As it was discussed in the paper, M2M environment is capable of implementing key features of such an approach, and given case study showed how it is possible to reach state of information superiority in the e-Health domain. Future work includes further research on the topic of information management in M2M networks and possible development of the network-centric information grid framework using the multi-agent software paradigm.

References

1. Lawton, G.: Machine-to-Machine Technology Gears Up for Growth. IEEE Computer 37(9), 12–15 (2004)
2. Smart Devices and Services Connected by CDMA 2000, Harbor Research, Inc. (2010)
3. M2M: Growth Opportunities for MNOs in developed Markets (Sample Pages), Mobile Market Development Ltd. (2010)
4. Alendal, M.: Operators need an ecosystem to support 50 billion connections. Ericsson Review (3) (2010)
5. Cox, A.: M2M: Connecting the devices. M2M & Embedded Strategies, Telematics, CE, mHealth, Metering & Smart Buildings 2011-2016, Juniper Research (2011)
6. ETSI M2M Technical Committee, http://www.etsi.org/Website/Technologies/M2M.aspx
7. 3GPP Technical Specification 22.368, v11.3.0 (2011), http://www.3gpp.org/ftp/Specs/html-info/22368.htm
8. Lien, S.-Y., Chen, K.-W., Lin, Y.: Towards Ubiquitous Massive Accesses in 3GPP Machine-to-Machine Communications. IEEE Communications Magazine 49(4), 66–74 (2011)
9. Chang, K., Soong, A., Tseng, M., Xiang, Z.: Global Wireless Machine-to-Machine Standardization. IEEE Internet Computing 15(2), 64–69 (2011)
10. ETSI, http://www.etsi.org/website/homepage.aspx
11. 3GPP, http://www.3gpp.org/About-3GPP
12. Dressler, F.: Network-centric Actuation Control in Sensor/Actuator Networks based on Bio-inspired Technologies. In: Mobile Adhoc and Sensor Systems (MASS), pp. 680–684 (2006)
13. Cebrowski, A.K., Garstka, J.J.: Network-centric warfare: Its origin and future. U.S. Naval Institute Proceedings Magazine, 28–35 (1998)
14. NIS23, Net-Centric Information & Integration Services for Security System (2009)
15. Alberts, D.S., Garstka, J.J., Stein, F.P.: Network centric warfare: Developing and leveraging information superiority, 2nd edn. CCRP (2001)
16. Networked Field Services, ClickSoftware (2010)
17. Network Centric Warfare, Department of Defense Report to Congress (2001)
18. Wang. N., Schmidt, D.C., van't Hag, H., Corsaro, A.: Toward an adaptive Data Distribution Service for Dynamic Large-Scale Network-Centric Operation and Warfare. PrismTech Corporation (2008)
19. Wesensten, N.J., Belenky, G., Balkin, T.J.: Cognitive Readiness in Network-Centric Operations. Parameters 35(1), 94–105 (2005)
20. Bay, J.S.: Disruptive Effects of Net-Centricity on Command and Control. In: 13th ICCRTS "C2 for Complex Endeavors" (2008)
21. Groh, J.L.: Network-Centric Warfare: Leveraging the Power of Information. In: USAWC Guide to National Security Issues. Theory of War and Strategy, vol. 1, pp. 323–338 (2008)
22. von Lubitz, D., Wickramasinghe, N.: Healthcare Network Centric Operations: The Confluence of E-Health and E-Government. In: Global E-Government: Theory, Applications and Benchmarking, pp. 127–147 (2007)
23. ETSI Technical Report 102.742: Use cases of M2M applications for eHealth, v0.4.1 (2011), http://docbox.etsi.org/M2M/Open/Latest_Drafts/00005v041.pdf
24. Schmidt, D.C., Corsaro, A., van't Hag, H.: Addressing the Challenges of Tactical Information Management in Net-Centric Systems with DDS. PrismTech Corporation (2008)

Deployment of Trust Management System in Environment of e-Commerce

Eva Zupančič and Denis Trček

Faculty of Computer and Information Science,
Tržaška 25, SI-1000 Ljubljana, Slovenia
{eva.zupancic,denis.trcek}@fri.uni-lj.si

Abstract. Internet has become an important part of our everyday life, as it has created new opportunities for business and offered new ways of social gathering. A very critical topic concerning decision making by online transactions is trust issue. This paper presents a formal representation of a sample e-commerce environment with communicating agents, including their properties and assets. Further, we describe rules of an investment game that agents are playing. We propose two different scenarios. In the first one, agents' investment decisions are ad hoc. In the second one, there is trust management system deployed that affects the agents' decisions. Based on developed simulation tool, we investigate the proposed scenarios. We show that with introducing trust management system agents filter out bad agents and only good ones prosper. We also discuss other evident effects of trust management system on agents and their final investments values based on simulations results.

Keywords: trust management, e-commerce, multi-agent systems, simulations.

1 Introduction

Internet has become a business medium and there are growing number of participants engaging in electronic commerce [3]. Conducting business over the Internet is cheaper and faster than making it in "traditional" way. On the other hand, it means trading with unknown entities. A critical issue concerning online interactions is trustworthiness of participating entities. Trust and reputation management systems should be deployed to distinguish between *bad* and *good* entities and mitigate risk involved in transactions, consequently.

In this paper, we present multi-agent system representing e-commerce environment. The emphasis is on trust relationships between participating agents and rules of modified investment game played by them. Further, we propose two different scenarios: in the first one, agents have no decision support by making transactions with other agents. In the second one, we deploy trust management system. Based on simulations, we compare the scenarios and test if there are some benefits with deployed trust management system.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 484–493, 2012.

The rest of the paper is structured as follows. The following section gives an overview of the field. Section 3 presents a formal model of e-commerce environment, description of investment game and proposed scenarios for simulations. Next, we present and analyse received results. Section 5 concludes the article.

2 Related Work

In recent years, different trust and reputation models have been presented [15,10,7,12,5,9,11]. Frequently used methods for trust computation are probability calculus and its derivatives, game theory approaches and fuzzy logic.

The mathematical basis for probability trust models is Bayes' theorem. The calculation of the posterior probability of hypothesis h after observing evidence e (i.e. $p(h|e) = \frac{p(e|h) \cdot p(h)}{p(e)}$) is the theoretical foundation of trust models implemented by Wang and Vassilieva [15] and Mui, *et al.* [10].

A derivative of probability calculus models is Jøsang's subjective logic [7]. It models trust in x as $\omega(x) = (b(x), d(x), u(x))$, where $b(x)$ means an observer's total belief that a particular state is true, $d(x)$ means total belief that a particular state is not true and $u(x)$ represents the uncertainty regarding given state. Additionally, Jøsang proposes various operators for combining trust opinions, such as consensus, recommendation, etc.

In game theory based trust models, interactions among agents are represented as a game. Tennenholtz [12] and Harish, *et al.,* [5] presented game theoretic models that suggest different strategies agents should apply.

Manhala [9] and Sabater and Sierra [11] proposed trust models using linguistic variables and they used fuzzy logic to operate with them.

The more extensive overview of current status of trust in computing environment could be found in [8,4,10].

3 Methods

3.1 Formalization of E-commerce Environment

A formalization of trust is based on qualitative assessment dynamics (QAD) [13], which models trust with qualitative terms instead of numeric values. Experiments in [13] show that qualitative assessment of trust is more appropriate than quantitative. Fruther, QAD emphasize the socio-cognitive nature of trust phenomenon with introduction of operators that model behavior of agents.

An environment of e-commerce consists of society of agents that make transactions with each other. In order to describe such e-environment in a formal way, we introduce the following definitions.

Definition 1. *A society is a set of communicating agents and certain trust relations between them and is denoted as* $S = (A, \mathbf{T})$.

An agent is described with 4-tuple $a = (id, m, h, op)$. The $id \in \mathbb{N}_0$ is a number of the agent and uniquely identifies it. In further sections, we denote agents with letters A, B, C, etc. which represent id numbers of agents. The m is amount of money, where $m \in \mathbb{R}$. The agents can increase or decrease their amount of money with making transactions. The $h \in [0, 1]$ is level of honesty and does not change over time. $h = 1$ means an agent is honest and $h = 0$ means dishonest agent. The value of h defines the agent's actions in transactions. The op presents an operator. It is an element of a set $\Psi \in \{\Uparrow, \Downarrow, \rightsquigarrow\}$, where symbols denote optimistic, pessimistic and centralistic operator. The operator is a function and it models an agent's trust relation towards other agents.

Definition 2. *Trust is a relationship between agents A and B. It is denoted by $\omega_{A,B}$, which means agent's A attitude towards agent B. Value of the relationship is taken from assessment set $\Omega = \{\mathsf{T}, \mathsf{PT}, \mathsf{U}, \mathsf{PDT}, \mathsf{DT}\}$, which means trusted, partially trusted, undecided, partially distrusted, and distrusted relationship, respectively.*

Chang et al. [2] define a *trusting agent* as an entity who has faith of belief in another entity in a given context and at given time slot. Further, they define a *trusted agent* as an entity in whom faith or belief has been placed by another entity in a given context and at given time slot. In terms of QAD, trusted agent is an agent that trusting agent has defined, or known, trust relationship towards him, even it is *distrusted*. In definition 2, agent A represents trusting agent and agent B represents trusted agent.

The assessment set consists of qualitative values, $\Omega = \{\mathsf{T}, \mathsf{PT}, \mathsf{U}, \mathsf{PDT}, \mathsf{DT}\}$, which describe trusted, partially trusted, undecided, partially distrusted and distrusted trust relationship between agents, respectively. If trust relationship between two agents is not defined or unknown, it is denoted by "–". The assessment values from Ω are arranged on an ordinal scale in the following way: $\mathsf{DT} < \mathsf{PDT} < \mathsf{U} < \mathsf{PT} < \mathsf{T}$.

This implies that values are assigned to relationships based on their ranking with respect to one other. While we know that PDT indicates more trusted relation than DT and PT indicates more trusted relation than U, there is no implication that the improvement from DT to PDT means the same "amount" as the improvement from U to PT. All we know is that there are 5 categories and "for how many levels" one category differs from another: $\mathsf{d}(\omega_1, \omega_2) = |\omega_2 - \omega_1|_\Omega$. For example, $\mathsf{d}(\mathsf{DT}, \mathsf{T}) = 4$ and $\mathsf{d}(\mathsf{PT}, \mathsf{U}) = 1$.

Further, we define two operations to work with values from Ω.

Definition 3. *Let $+_\Omega$ denote Ω-addition operator that is applied to trust value ω^- and results in ω^+. Result of the Ω-addition operation $\omega^+ = +_\Omega(\omega^-)$ is the value's ω^- next right value on the trust scale. If ω^- is the most-right value, than $\omega^+ = \omega^-$.*

Definition 4. *Let $-_\Omega$ denote Ω-subtraction operator that is applied to trust value ω^- and results in ω^+. Result of the Ω-subtraction operation $\omega^+ = -_\Omega(\omega^-)$ is the value's ω^- next left value on the trust scale. If ω^- is the most-left value, than $\omega^+ = \omega^-$.*

We define middle, lower-middle and upper-middle values of two ω values by means of Ω-addition and Ω-subtraction operations. The middle value of ω_1 and ω_2 exists only if the distance between them is even. In opposite case, we can only define lower-middle and upper-middle values.

Definition 5. *Let* $\omega_1 \leq \omega_2$*. If* $\mathtt{d}(\omega_1, \omega_2) = 2k, \forall k \in \mathbb{Z}$ *then the middle value* ω_{mid} *of* ω_1 *and* ω_2 *is computed according to the following rule:* $+^*_\Omega(\omega_1) = -^*_\Omega(\omega_2) = \omega_{mid}(\omega_1, \omega_2)$*. Ω-addition should be applied to* ω_1 *value and Ω-subtraction should be applied to* ω_2 *value so many times until they result in the same value on the trust scale. In the case of* $\omega_1 = \omega_2$ *the middle value equals* $\omega_{mid}(\omega_1, \omega_2) = \omega_1 = \omega_2$*.*

Definition 6. *Let* $\omega_1 < \omega_2$*. If* $\mathtt{d}(\omega_1, \omega_2) = 2k+1, \forall k \in \mathbb{Z}$ *then the lower-middle value* $\omega_{mid_\vee}$ *and the upper-middle value* $\omega_{mid_\wedge}$ *of* ω_1 *and* ω_2 *is computed according to the following rule:* $\omega_{mid_\vee}(\omega_1, \omega_2) = +^*_\Omega(\omega_1) < -^*_\Omega(\omega_2) = \omega_{mid_\wedge}(\omega_1, \omega_2)$*. Ω-addition should be applied to* ω_1 *value and Ω-subtraction should be applied to* ω_2 *value so many times until result of Ω-additions on* ω_1 *is still to left to the result of Ω-subtractions on* ω_2 *on the trust scale. In the case of* $\mathtt{d}(\omega_1, \omega_2) = 1$ *the lower-middle value equals* $\omega_{mid_\vee}(\omega_1, \omega_2) = \omega_1$ *and the upper-middle value equals* $\omega_{mid_\wedge}(\omega_1, \omega_2) = \omega_2$*.*

To simplify further definitions and algorithms we introduce the following function:

$$\mathtt{mid}(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } \mathtt{d}(\omega_1, \omega_2) = 2k, \forall k \in \mathbb{Z}; \\ 2 & \text{else.} \end{cases}$$

If distance between two ω values is even then there is only one middle value, ω_{mid}, and the result of `mid` function is 1. Otherwise, there are two middle values, $\omega_{mid_\vee}$ and $\omega_{mid_\wedge}$, and the result of `mid` function is 2.

The `d`, `mid`, $\mathtt{mid}_\vee$ and $\mathtt{mid}_\wedge$ definitions will be later used to define operators (see Eq. 1, 2, 3).

The trust values among agents are stored in a trust matrix $\mathbf{T}$. In a society with n agents, a general form of trust matrix $\mathbf{T}$ is as follows:

$$\mathbf{T} = \begin{bmatrix} \omega_{1,1} & \omega_{1,2} & \cdots & \omega_{1,n} \\ \omega_{2,1} & \omega_{2,2} & \cdots & \omega_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{n,1} & \omega_{n,2} & \cdots & \omega_{n,n} \end{bmatrix}.$$

The agents are arranged on a two-dimensional square grid that wraps around into shape of torus. Fig. 1 shows an example of such arrangement with 25 agents. Numbers represent *id* numbers of the agents.

Definition 7. *An agent's A neighborhood is a set of agents which are positioned north, north-east, east, south-east, south, south-west, west, and north-west of A; it is denoted as*

$$\mathcal{N}_A = \{a_{N(A)}, a_{NE(A)}, a_{E(A)}, a_{SE(A)}, a_{S(A)}, a_{SW(A)}, a_{W(A)}, a_{NW(A)}\}.$$

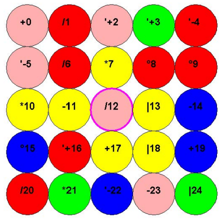


Fig. 1. Agents are arranged on square grid

Fig. 1 is given to illustrate it. For example, neighbourhood of agent 12 is clearly seen and it holds $\mathcal{N}_{12} = \{7, 8, 13, 18, 17, 16, 11, 6\}$.

Definition 8. $\omega_{mode[\mathcal{N}_A(B)]}$ *is a value that occurs the most frequently in set* $\{\omega_{N(A),B}, \omega_{NE(A),B}, \omega_{E(A),B}, \omega_{SE(A),B}, \omega_{S(A),B}, \omega_{SW(A),B}, \omega_{W(A),B}, \omega_{NW(A),B}\}$. *If there is more than one mode, then we consider the value with the highest priority. The priorities are defined in the following way:* 1. U*,* 2. PT*,* 3. PDT*,* 4. T*,* 5. DT*.*

It represents the most frequent trust value of agent's A neighbors towards agent B.

In a trust matrix $\mathbf{T}$, row K represents agent's K trust towards other agents, and column K represents society trust towards particular agent K. Considering only agent's A neighbors trust towards particular agent K results in a neighborhood trust sub-vector, i.e. $\mathbf{T}_{\mathcal{N}_A(K)} = (\omega_{N(A),K}, \omega_{NE(A),K}, ..., \omega_{NW(A),K})$.

The neighborhood could be expanded or interpreted as social, business or family connections [14,11]. A plane with square grid wraps around into shape of torus, such that the agents that are positioned in the very right column are adjacent to the agents that are positioned in the very left column, as an example. The shape of torus enables equivalent position to all agents. Each agent has the same number of neighbors and the same possibility to spread opinion and impact on the society.

3.2 Investment Game

There are many different theoretic games to study trust. The investment game (also known as "trust game") [1] has been widely used in the areas of research, such as economic psychology and behavioural economy [6]. It is suitable for modelling e-commerce systems, as it includes aspects of trust, risk and uncertain payments.

The investmet game is played between agent A and agent B. The agent A has a role of an investor and the agent B has a role of a partner. The investor A sends M_A money to the partner B, which receives tripled amount of sent money, i.e. $3M_A$. In second phase, the agent B must decide how much money to return. In our variation of the investment game, it depends on agent's honesty and it sends back $h_b(3M_A)$, where h_b is honesty level of B. More honest agent sends back more money.

The payoff for each agent are as follows:

$$Payoff_A(M_A, h_b) = -M_A + h_b(3M_A), \qquad Payoff_B(M_A, h_b) = 3M_A - h_b(3M_A).$$

The investor A values the investment with B based on payoff as follows:

$$\omega_{p_{(A,B)}}(Payoff_A(M_A, h_b)) = \begin{cases} \mathsf{DT} & \text{for } Payoff_A(M_A, h_b) < 0, \\ \mathsf{PDT} & \text{for } 0 \leq Payoff_A(M_A, h_b) < \frac{M_A}{2}, \\ \mathsf{U} & \text{for } \frac{M_A}{2} \leq Payoff_A(M_A, h_b) < M_A, \\ \mathsf{PT} & \text{for } M_A \leq Payoff_A(M_A, h_b) < 3\frac{M_A}{2}, \\ \mathsf{T} & \text{for } Payoff_A(M_A, h_b) \geq 3\frac{M_A}{2}. \end{cases}$$

Agent A assesses the investment with values from the set Ω. If an investment is assessed with DT, it means distrusted investment with loss. Similarly, trusted investment (i.e. $\omega_{p_{(A,B)}} = \mathsf{T}$) means high profit.

The agent A decides on further transactions with the partner B based on $\omega_{p_{(A,B)}}$ value, among other factors.

3.3 Scenarios

We propose two different scenarios for playing investment games. In the first one, an agent A *always* makes a transaction with an agent B. After the transaction, agents update the state of their money as follows: $m_A^+ = m_A^- + Payoff_A$ and $m_B^+ = m_B^- + Payoff_B$, where the superscript "$-$" denotes value before investment transaction and the superscript "+" denotes value after the transaction.

In the second scenario, agents A and B make interaction first and they check trust relationship values between them. If A trusts B and vice versa, i.e. $\omega_{A,B} \geq$ *threshold* and $\omega_{B,A} \geq$ *threshold*, then they proceed transaction and A sends money to B.

After the transaction, agents update the state of their money and investor values the investment. Additionally, trust relationship value is updated after the interaction – whether A has made transaction with B or not. Trust relationship value is updated as follows: $\omega_{A,B}^+ = op_A(\omega_{A,B}^-, \omega_{p_{(A,B)}}, \omega_{mode[\mathcal{N}_A(B)]})$.

The operator $op_A \in \Psi$ means function assigned to agent A and mappings are defined in the following way:

$$\Uparrow_A: \quad \mathtt{max}(\omega_{A,B}^-, \omega_{p_{(A,B)}}, \omega_{mode[\mathcal{N}_A(B)]}) \rightarrow \omega_{A,B}^+ \tag{1}$$

$$\Downarrow_A: \quad \mathtt{min}(\omega_{A,B}^-, \omega_{p_{(A,B)}}, \omega_{mode[\mathcal{N}_A(B)]}) \rightarrow \omega_{A,B}^+ \tag{2}$$

$$\rightsquigarrow_A: \begin{cases} \mathtt{mid}(\omega_{A,B}^-, \omega_{p_{(A,B)}}) \rightarrow \omega_{A,B}^+ & \text{if } \mathtt{d}(\omega_{A,B}^-, \omega_{p_{(A,B)}}) = 2k, \forall k \in \mathbb{Z}, \\ \mathtt{mid}_\wedge(\omega_{A,B}^-, \omega_{p_{(A,B)}}) \rightarrow \omega_{A,B}^+ & \text{else if } \omega_{mode[\mathcal{N}_A(B)]} \geq \mathsf{U}, \\ \mathtt{mid}_\vee(\omega_{A,B}^-, \omega_{p_{(A,B)}}) \rightarrow \omega_{A,B}^+ & \text{else.} \end{cases} \tag{3}$$

An agent A compares trust value towards an agent B as it was before interaction (i.e. $\omega^{-}_{A,B}$), trust value that reflexes the output of the interaction (i.e. $\omega_{p_{(A,B)}}$) and the trust values of its neighbours towards the agent B. The agent with optimistic operator ($\Uparrow$) chooses the highest of those values, while the agent with pessimistic operator ($\Downarrow$) chooses the lowest of those values. The agent with centralist operator (takes the middle value between trust value before and after interaction. If the distance between these two values is odd, then the agent A chooses the *upper-middle* value if its neighbours have good opinions of the agent B, and it chooses *lower-middle* value otherwise.

4 Results

Based on the formal description of e-commerce environment and rules of investment game presented in previous sections, we have implemented simulation tool *Zaupnik*.

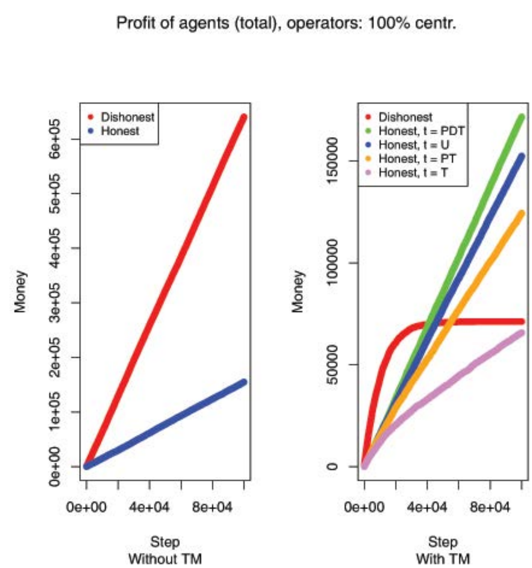


Fig. 2. Effect of TM deployment on total profit of dishonest and honest agents

The simulations are run with total $n = 100$ agents and uniform distribution of honesty $H = [0, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n}{n} = 1]$. In each step, two agents are randomly chosen and make investment game transaction (if conditions are satisfied). Each agent invests the same amount of money: $M_A = 10$. We assume that agents can have negative amount of money, too. At the beginning, they have same amount of money. As there are no money limitations in present simulations, we could choose any constant as initial amount of money. For the sake of simplicity we set it to 0.

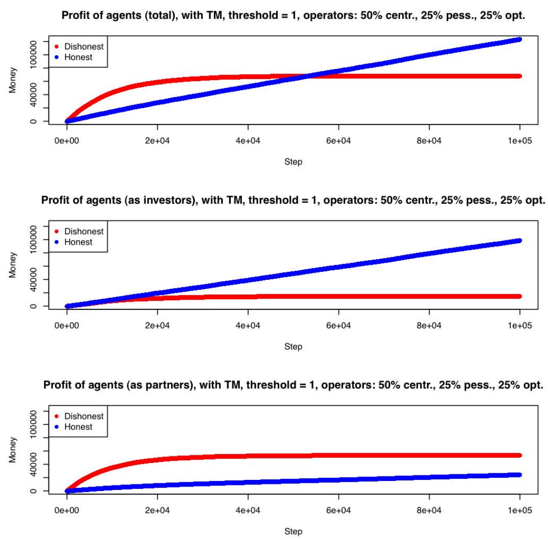


Fig. 3. Profit of agents according to their role in the investment game in the system with TM deployed

4.1 The First Experiment

Firstly, we compare how much money agents earn over time. We divided agents into two groups: agents with $h \geq 0.8$ are referred to as *honest* and agents with $h \leq 0.2$ are referred to as *dishonest*. In scenario without trust management system, dishonest agents get more money as honest agents. Dishonest partners get tripled amount of money sent by investors, but they send back at most 20% of received money. We prevent investors to make transactions with such partner with deployment of trust management system.[1] In scenario with deployed trust management system, honest agents get more money as dishonest agents over time. Fig. 2 shows the comparison of dishonest and honest agents' profit in two different scenarios. Fig. 3 shows the agents' profit according to being earned as investor, partner or in total.

4.2 The Second Experiment

Secondly, we examine the effect of honesty and different operators on profit. In the scenario without trust management system, the most dishonest agent, i.e. $h_A = 0$, have the highest profit as it increases its budget for tripled amount of investor's money in every transaction. In the scenario with deployed trust management system, agents with highest profit are those that return *just enough*

[1] Trust management system refers to decision support system based on trust perspectives.

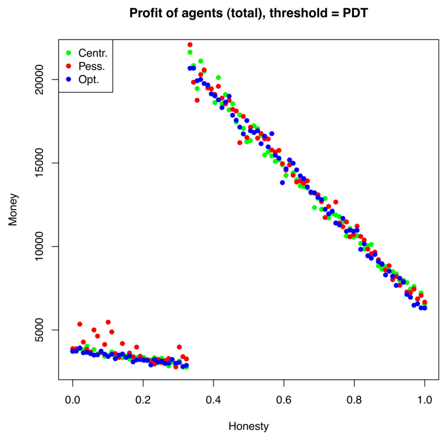


Fig. 4. Total profit of agents depending on operators

money to be evaluated as trusted enough for further transactions. For instance, a transaction is assessed as partially distrusted (PDT) if investor's payoff is at least 0. Such payoff is achieved when partner's honesty level is $h_B = \frac{1}{3}$ and its payoff is $Payoff_B(M_A, \frac{1}{3}) = 2M_A$. Hence, if threshold for entering the transaction is PDT, then agents with $h_B = \frac{1}{3}$ have the highest profit. The profit of agents in dependence of their honesty level is shown in Fig. 4.

The effect of operators on agents' final budget is insignificant In trust management systems with low threshold for transaction, e.g. $t =$ PDT, dishonest agents with pessimistic operators are better off. Agents with pessimistic operators lean to assign distrusted trust relationship values. The possibility that transaction is done is relatively high because of tolerant threshold, which results in high profits for dishonest agents. The profit of agents in dependence of their operators is also shown in Fig. 4.

5 Discussion

In this paper we presented formalization of e-commerce environment, together with rules of transactions between agents that are modified investment game rules. We implemented a simulation tool to simulate different scenarios of playing investment games transactions between agents.

Results show that deployment of trust management system into context of e-commerce could protect honest agents against dishonest ones. In such environment, the honest agents benefit in long term period, i.e. after (bad) experiences with dishonest agents are collected.

In proposed scenarios with transactions based on investment game rules, the final profit of agent depends on its honesty. In environment without trust management system, the most dishonest agents have the highest profit. With deployment of trust management system, we prevent dishonest agent from high profits to some extent. Differently, the agent's operator does not have significant impact on agent's final budget.

In future work, we shall propose new operators and improve trust management mechanisms. With extensive simulations we will discover their effects on society and different group of agents and examine possibilities to manage trust in context of e-commerce.

References

1. Berg, J., Dickhaut, J., McCabe, K.: Trust, reciprocity, and social history. Games and Economic Behavior 10(1), 122–142 (1995)
2. Chang, E., Dillon, T.S., Hussain, F.K.: Trust and reputation for service-oriented environments: technologies for building business intelligence and consumer confidence. John Wiley & Sons (2006)
3. Eastin, M.S.: Diffusion of e-commerce: an analysis of the adoption of four e-commerce activities. Telematics and Informatics 19(3), 251–267 (2002)
4. Grandison, T., Sloman, M.: A survey of trust in internet applications. IEEE Communications Surveys & Tutorials 3(4), 2–16 (2009)
5. Harish, M., Anandavelu, N., Anbalagan, N., Mahalakshmi, G.S., Geetha, T.V.: Design and Analysis of a Game Theoretic Model for P2P Trust Management. In: Janowski, T., Mohanty, H. (eds.) ICDCIT 2007. LNCS, vol. 4882, pp. 110–115. Springer, Heidelberg (2007)
6. Johnson, N.D., Mislin, A.A.: Trust games: A meta-analysis. Journal of Economic Psychology 32(5), 865–889 (2011)
7. Jøsang, A.: A logic for uncertain probabilities. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 9(3) (2001)
8. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems 43(2), 618–644 (2007)
9. Manchala, D.W.: Trust metrics, models and protocols for electronic commerce transactions. In: Proceedings 18th International Conference on Distributed Computing Systems, 1998, pp. 312–321. IEEE (2002)
10. Mui, L., Mohtashemi, M., Halberstad, A.: A computational model of trust and reputation. In: Proceedings of the 35th Annual Hawaii International Conference on System Sciences, HICSS, pp. 2431–2439. IEEE (2002)
11. Sabater, J., Sierra, C.: Reputation and Social Network Analysis in Multi-Agent Systems (2002)
12. Tennenholtz, M.: Game-theoretic recommendations: some progress in an uphill battle. In: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, vol. 1, pp. 10–16. International Foundation for Autonomous Agents and Multiagent Systems (2008)
13. Trček, D.: A formal apparatus for modeling trust in computing environments. Mathematical and Computer Modelling 49(1-2), 226–233 (2009)
14. Walter, F.E., Battiston, S., Schweitzer, F.: A model of a trust-based recommendation system on a social network. Autonomous Agents and Multi-Agent Systems 16(1), 57–74 (2008)
15. Wang, Y., Vassileva, J.: Bayesian Network Trust Model in Peer-to-Peer Networks. In: Moro, G., Sartori, C., Singh, M.P. (eds.) AP2PC 2003. LNCS (LNAI), vol. 2872, pp. 23–34. Springer, Heidelberg (2004)

SAT-Based Bounded Model Checking for Deontic Interleaved Interpreted Systems*

Bożena Woźna-Szcześniak and Andrzej Zbrzezny

IMCS, Jan Długosz University, Al. Armii Krajowej 13/15, 42-200 Częstochowa, Poland
{b.wozna,a.zbrzezny}@ajd.czest.pl

Abstract. We propose a bounded model checking (BMC) method for the verification of multi-agent systems' (MASs). The MASs are modelled by deontic interleaved interpreted systems, and specifications are expressed in the logic RTECTLKD. The verification approach is based on the state of the art solutions to BMC, one of the mainstream approaches in verification of reactive systems. We test our results on a typical communication scenario: train controller problem with faults.

1 Introduction

Agents are rational and sophisticated entities that act autonomously on behalf of their users, across open and distributed environments, to solve a growing number of complex problems. *A multi-agent system* (MAS) [14] is a loosely united network of agents that interact (communicate, coordinate, cooperate, etc.) to solve problems that are beyond the individual capacities or knowledge of a single agent. *Deontic interpreted systems* (DIS), a deontic extension of interpreted systems [4], were defined in [8] to represent and reason about epistemic and correct functioning behaviour of MASs. They provide a semantics based on the computation states of the agents, on which it is possible to interpret a modality $\mathcal{O}_i\phi$, representing the fact "in all correct functioning executions of agent i, ϕ holds", as well as a traditional epistemic modalities and temporal operators. *Deontic interleaved interpreted systems* (DIIS) are a deontic extension of the formalism of interleaved interpreted systems [7]. We introduce them since they allow for the distinction between correct (or ideal, normative, etc.) and incorrect states, and they enable more efficient verification of MASs that are not so loosely coupled.

Model checking [2] is one of the mainstream techniques whose aim is to provide an algorithm determining whether an abstract model - representing, for example, a software project - satisfies a formal specification expressed as a modal formula. Moreover, if the property does not hold, the method identifies a counterexample execution that shows the source of the problem. The practical applicability of model checking in MASs settings requires the development of sophisticated means of coping with what is known as the state explosion problem. To avoid this problem a number of approaches have been developed, including BDD-based bounded [5,9] and unbounded [13,12] model checking, SAT-based bounded [10,11,15] and unbounded [6] model checking.

* Partly supported by National Science Center under the grant No. 2011/01/B/ST6/05317.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 494–503, 2012.

The RTCTLKD language is an epistemic and deontic extension of RTCTL [3], which allows for the representation of the quantitative temporal evolution of epistemic states of the agents, as well as their correct and incorrect functioning behaviour.

In past research we have provided a theoretical underpinnings of a bounded model checking (BMC) algorithm for DIS and an existential part of CTLKD (ECTLKD) [15]. However, the method have not been implemented and experimentally evaluated. Moreover, it was not tailored to the DIISs settings, and it was not based on the state-of-the art BMC method for ECTL [17], which uses a reduced number of paths, what results in significantly smaller and less complicated propositional formulae that encode the ECTLKD properties. In this paper we provide a new SAT-based BMC technique for the existential part of RTCTLKD (thus, for ECTLKD as well) by means of which we can automatically verify not only epistemic and temporal properties but also deontic and quantitative temporal properties that express compliance of a MAS, modelled by DIIS, with respect to specifications.

The structure of the paper is as follows. In Section 2 we shortly introduce DIISs, the RTCTLKD language together with its existential (RTECTLKD) and universal fragments (RTACTLKD), unbounded and bounded semantics. In Section 3 we define a BMC method for RTECTLKD. In Section 4 we present performance evaluation of our newly developed SAT-based BMC algorithm and we conclude the paper.

2 Preliminaries

DIIS. We assume that a MAS consists of n agents, and by $Ag = \{1, \ldots, n\}$ we denote the non-empty set of agents; note that we do not consider the environment component. This may be added with no technical difficulty at the price of heavier notation. We assume that each agent $c \in Ag$ is in some particular local state at a given point in time, and that a set L_c of local states for agent $c \in Ag$ is non-empty and finite (this is required by the model checking algorithms). We assume that for each agent $c \in Ag$, its set L_c can be partitioned into *faultless (green)* and *faulty (red)* states. For n agents and n mutually disjoint and non-empty sets $\mathcal{G}_1, \ldots, \mathcal{G}_n$ we define the set G of all possible *global states* as the Cartesian product $L_1 \times \ldots \times L_n$, such that $L_1 \supseteq \mathcal{G}_1, \ldots, L_n \supseteq \mathcal{G}_n$. The set $\mathcal{G}_c$ is called the set of green states for agent c. The complement of $\mathcal{G}_c$ with respect to L_c (denoted by $\mathcal{R}_c$) is called the set of red states for the agent c. Note that $\mathcal{G}_c \cup \mathcal{R}_c = L_c$ for any agent c. Further, by $l_c(g)$ we denote the local component of agent $c \in Ag$ in a global state $g = (l_1, \ldots, l_n)$.

With each agent $c \in Ag$ we associate a finite set of *possible actions* Act_c such that a special "null" action (ϵ_c) belongs to Act_c; as it will be clear below the local state of agent c remains the same, if the null action is performed. We do not assume that the sets Act_c (for all $c \in Ag$) are disjoint. Next, with each agent $c \in Ag$ we associate a protocol that defines rules, according to which actions may be performed in each local state. The protocol for agent $c \in Ag$ is a function $P_c : L_c \to 2^{Act_c}$ such that $\epsilon_c \in P_c(l)$ for any $l \in L_c$, i.e., we insist on the null action to be enabled at every local state. For each agent c, there is defined a (partial) evolution function $t_c : L_c \times Act_c \to L_c$ such that for each $l \in L_c$ and for each $a \in P_c(l)$ there exists $l' \in L_c$ such that $t_c(l, a) = l'$; moreover, $t_c(l, \epsilon_c) = l$ for each $l \in L_c$. Note that the local evolution function considered here differs from the standard one (see [4]) by having the local action instead of the join action as the parameter. Further, we define the following sets $Act = \bigcup_{c \in Ag} Act_c$ and $Agent(a) = \{c \in Ag \mid a \in Act_c\}$.

We assumed that, in every state, agents evolve simultaneously. Thus the *global interleaved evolution function* $t : G \times Act_1 \times \cdots \times Act_n \to G$ is defined as follows: $t(g, a_1, \ldots, a_n) = g'$ iff there exists an action $a \in Act \setminus \{\epsilon_1, \ldots, \epsilon_n\}$ such that for all $c \in Agent(a)$, $a_c = a$ and $t_c(l_c(g), a) = l_c(g')$, and for all $c \in Ag \setminus Agent(a)$, $a_c = \epsilon_c$ and $t_c(l_c(g), a_c) = l_c(g)$. In brief we write the above as $g \xrightarrow{a} g'$.

Now, for a given set of agents Ag and a set of propositional variables $\mathcal{PV}$, which can be either true or false, a *deontic interleaved interpreted system* is a tuple: $DIIS = (\iota, < L_c, \mathcal{G}_c, Act_c, P_c, t_c >_{c \in Ag}, \mathcal{V})$, where $\iota \in G$ is an initial global state, and $\mathcal{V} : G \to 2^{\mathcal{PV}}$ is a valuation function. With such a DIIS it is possible to associate a *model* $M = (\iota, S, T, \{\sim_c\}_{c \in Ag}, \{\bowtie_c\}_{c \in Ag}, \mathcal{V})$, where ι is the initial global state; $S \subseteq G$ is a set of reachable global states that is generated from ι by using the global interleaved evolution functions t; $T \subseteq S \times S$ is a global transition (temporal) relation on S defined by: sTs' iff there exists an action $a \in Act \setminus \{\epsilon_1, \ldots, \epsilon_n\}$ such that $s \xrightarrow{a} s'$. We assume that the relation is total, i.e., for any $s \in S$ there exists an $a \in Act \setminus \{\epsilon_1, \ldots, \epsilon_n\}$ such that $s \xrightarrow{a} s'$ for some $s' \in S$; $\sim_c \subseteq S \times S$ is an indistinguishability relation for agent c defined by: $s \sim_c s'$ iff $l_c(s') = l_c(s)$; $\bowtie_c \subseteq S \times S$ is a deontic relation for agent c defined by: $s \bowtie_c s'$ iff $l_c(s') \in \mathcal{G}_c$; $\mathcal{V} : S \to 2^{\mathcal{PV}}$ is the valuation function of $DIIS$ restricted to the set S. $\mathcal{V}$ assigns to each state a set of propositional variables that are assumed to be true at that state.

Syntax of RTCTLKD. Let $p \in \mathcal{PV}$, $c \in Ag$, $\Gamma \subseteq Ag$, and I be an interval in $\mathbb{N} = \{0, 1, 2, \ldots\}$ of the form: $[a, b)$ and $[a, \infty)$, for $a, b \in \mathbb{N}$; note that the remaining forms of intervals can be defined by means of $[a, b)$ and $[a, \infty)$. Hereafter, let *left*(I) denote the left end of the interval I, and $right(I)$ the right end of the interval I. The language RTCTLKD is defined by the following grammar:

$$\varphi := \mathbf{true} \mid \mathbf{false} \mid p \mid \neg\alpha \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathrm{EX}\varphi \mid \mathrm{E}(\varphi \mathrm{U}_I \varphi) \mid \mathrm{EG}_I\varphi \mid$$
$$\overline{\mathrm{K}}_c\varphi \mid \overline{\mathrm{D}}_\Gamma\varphi \mid \overline{\mathrm{E}}_\Gamma\varphi \mid \overline{\mathrm{C}}_\Gamma\varphi \mid \overline{\mathcal{O}}_c\alpha \mid \underline{\widehat{\mathrm{K}}}_c^d\alpha$$

The derived basic modalities are defined as follows: $\mathrm{E}(\alpha \mathrm{R}_I \beta) \stackrel{def}{=} \mathrm{E}(\beta \mathrm{U}_I(\alpha \wedge \beta)) \vee \mathrm{EG}_I\beta$, $\mathrm{EF}_I\alpha \stackrel{def}{=} \mathrm{E}(\mathbf{true}\mathrm{U}_I\alpha)$, $\mathrm{AX}\alpha \stackrel{def}{=} \neg\mathrm{EX}\neg\alpha$, $\mathrm{AF}\alpha \stackrel{def}{=} \neg\mathrm{EG}\neg\alpha$, $\mathrm{A}(\alpha\mathrm{R}\beta) \stackrel{def}{=} \neg\mathrm{E}(\neg\alpha\mathrm{U}\neg\beta)$, $\mathrm{AG}\alpha \stackrel{def}{=} \neg\mathrm{EF}\neg\alpha$, $\mathcal{O}_c\alpha \stackrel{def}{=} \neg\overline{\mathcal{O}}_c\neg\alpha$, $\mathrm{K}_c\alpha \stackrel{def}{=} \neg\overline{\mathrm{K}}_c\neg\alpha$, $\widehat{\mathrm{K}}_c^d\alpha \stackrel{def}{=} \neg\underline{\widehat{\mathrm{K}}}_c^d\neg\alpha$, $\mathrm{D}_\Gamma\varphi \stackrel{def}{=} \neg\overline{\mathrm{D}}_\Gamma\neg\alpha$, $\mathrm{E}_\Gamma\varphi \stackrel{def}{=} \neg\overline{\mathrm{E}}_\Gamma\neg\alpha$, $\mathrm{C}_\Gamma\varphi \stackrel{def}{=} \neg\overline{\mathrm{C}}_\Gamma\neg\alpha$, where $c, d \in \mathcal{AG}$, and $\Gamma \subseteq \mathcal{AG}$. Intuitively, E and A mean, resp., there exists a computation, and for all the computations, U_I and G_I are the operators, resp., for "bounded until" and "bounded always". $\overline{\mathrm{K}}_c$ is the operator dual for the standard epistemic modality K_c ("agent c knows"), so $\overline{\mathrm{K}}_c\alpha$ is read as "agent c does not know whether or not α holds". Similarly, the modalities $\overline{\mathrm{D}}_\Gamma, \overline{\mathrm{E}}_\Gamma, \overline{\mathrm{C}}_\Gamma$ are the dual operators for $\mathrm{D}_\Gamma, \mathrm{E}_\Gamma, \mathrm{C}_\Gamma$ representing distributed knowledge in the group Γ, everyone in Γ knows, and common knowledge among agents in Γ. Further, we use the (double) indexed modal operators $\mathcal{O}_c$, $\overline{\mathcal{O}}_c$, $\widehat{\mathrm{K}}_c^d$ and $\underline{\widehat{\mathrm{K}}}_c^d$ to represent the *correctly functioning circumstances of agent* c. The formula $\mathcal{O}_c\alpha$ stands for "for all the states where agent c is functioning correctly, α holds". $\overline{\mathcal{O}}_c$ is the operator dual for the modality $\mathcal{O}_c$. The formula $\widehat{\mathrm{K}}_c^d\alpha$ is read as "agent c knows that α under the assumption that agent d is functioning correctly”. $\underline{\widehat{\mathrm{K}}}_c^d$ is the operator dual for the modality $\widehat{\mathrm{K}}_c^d$. We refer to [8] for a discussion of this notion; note that the operator $\overline{\mathcal{O}}_c$ is there referred to as $\mathcal{P}_c$.

Next, we define two sublogics of RTCTLKD. The first one is the existential fragment of RTCTLKD (RTECTLKD), defined by the following grammar: $\varphi ::= p \mid \neg p \mid \alpha \wedge \beta \mid \alpha \vee \beta \mid \mathrm{EX}\alpha \mid \mathrm{E}(\alpha \mathrm{U}_I \beta) \mid \mathrm{EG}_I \alpha \mid \overline{\mathrm{K}}_c \alpha \mid \overline{\mathrm{E}}_\Gamma \alpha \mid \overline{\mathrm{D}}_\Gamma \alpha \mid \overline{\mathrm{C}}_\Gamma \alpha \mid \overline{\mathcal{O}}_c \alpha \mid \underline{\widehat{\mathrm{K}}}_c^d \alpha$. The second on is the universal fragment of RTCTLKD (RTACTLKD), defined as: $\varphi ::= p \mid \neg p \mid \alpha \wedge \beta \mid \alpha \vee \beta \mid \mathrm{AX}\alpha \mid \mathrm{A}(\alpha \mathrm{R}_I \beta) \mid \mathrm{AF}_I \alpha \mid \mathrm{K}_c \alpha \mid \mathrm{E}_\Gamma \alpha \mid \mathrm{D}_\Gamma \alpha \mid \mathrm{C}_\Gamma \alpha \mid \mathcal{O}_c \alpha \mid \widehat{\mathrm{K}}_c^d \alpha$.

Semantics of RTCTLKD. Let M be a model for $DIIS$. A *path* in M is an infinite sequence $\pi = (s_0, s_1, \ldots)$ of states such that $(s_j, s_{j+1}) \in T$ for each $j \in \mathbb{N}$. For a path π, we take $\pi(j) = s_j$. By $\Pi(s)$ we denote the set of all the paths starting at $s \in S$. For the group epistemic modalities we define the following. If $\Gamma \subseteq Ag$, then $\sim_\Gamma^E \stackrel{def}{=} \bigcup_{c \in \Gamma} \sim_c$, $\sim_\Gamma^C \stackrel{def}{=} (\sim_\Gamma^E)^+$ (the transitive closure of $\sim_\Gamma^E$), and $\sim_\Gamma^D \stackrel{def}{=} \bigcap_{c \in \Gamma} \sim_c$. Given the above, the semantics of RTCTLKD is the following:

- $M, s \models \mathbf{true}$, • $M, s \not\models \mathbf{false}$, • $M, s \models p$ iff $p \in \mathcal{V}(s)$, • $M, s \models \neg\alpha$ iff $M, s \not\models \alpha$,
- $M, s \models \alpha \wedge \beta$ iff $M, s \models \alpha$ and $M, s \models \beta$,
- $M, s \models \alpha \vee \beta$ iff $M, s \models \alpha$ or $M, s \models \beta$,
- $M, s \models \mathrm{EX}\alpha$ iff $(\exists \pi \in \Pi(s))(M, \pi(1) \models \alpha)$,
- $M, s \models \mathrm{E}(\alpha \mathrm{U}_I \beta)$ iff $(\exists \pi \in \Pi(s))(\exists m \in I)[M, \pi(m) \models \beta$ and $(\forall j < m) M, \pi(j) \models \alpha]$,
- $M, s \models \mathrm{EG}_I \alpha$ iff $(\exists \pi \in \Pi(s))$ such that $(\forall m \in I)[M, \pi(m) \models \alpha]$,
- $M, s \models \overline{\mathrm{K}}_c \alpha$ iff $(\exists s' \in S)(s \sim_c s'$ and $M, s' \models \alpha)$,
- $M, s \models \overline{Y}_\Gamma \alpha$ iff $(\exists s' \in S)(s \sim_\Gamma^Y s'$ and $M, s' \models \alpha)$, where $Y \in \{\mathrm{D}, \mathrm{E}, \mathrm{C}\}$,
- $M, s \models \overline{\mathcal{O}}_c \alpha$ iff $(\exists s' \in S)(s \bowtie_c s'$ and $M, s' \models \alpha)$,
- $M, s \models \underline{\widehat{\mathrm{K}}}_c^d \alpha$ iff $(\exists s' \in S)(s \sim_c s'$ and $s \bowtie_d s'$ and $M, s' \models \alpha)$.

An RTCTLKD formula φ is *valid* in M (denoted $M \models \varphi$) iff $M, \iota \models \varphi$, i.e., φ is true at the initial state of the model M. The *model checking problem* asks whether $M \models \varphi$.

Bounded Semantics. The proposed bounded semantics is the backbone of the SAT-based BMC method for RTECTLKD, which is presented in the next section. As usual, we start by defining *k-paths* and *loops*.

Let M be a model for $DIIS$, $k \in \mathbb{N}$, and $0 \leq l \leq k$. A *k-path* is a finite sequence $\pi = (s_0, \ldots, s_k)$ of states such that $(s_j, s_{j+1}) \in T$ for each $0 \leq j < k$. A k-path π is a *loop* if $\pi(k) = \pi(l)$ for some $l < k$. By $\Pi_k(s)$ we denote the set of all the k-paths starting at s in M. Note that although every k-path π is finite, if it is a loop, then it generates the infinite path of the following form: $u \cdot v^\omega$ with $u = (\pi(0), \ldots, \pi(l))$ and $v = (\pi(l+1), \ldots, \pi(k))$. Further, since in the bounded semantics we consider finite prefixes of paths only, the satisfiability of all the temporal operators depends on whether a considered k-path is a loop. Thus, as customary, we introduce a function $loop : \bigcup_{s \in S} \Pi_k(s) \to 2^{\mathbb{N}}$, which identifies these k-paths that are loops. The function is defined as: $loop(\pi_k) = \{l \mid 0 \leq l < k$ and $\pi_k(l) = \pi_k(k)\}$.

Given the above, the bounded semantics of RTECTLKD is defined as follows. Let $M, s \models_k \alpha$ denotes that α is $k-$true at the state s of M. The relation $\models_k$ is defined inductively as follows:

- $M, s \models_k$ **true**, • $M, s \not\models_k$ **false**, • $M, s \models_k p$ *iff* $p \in \mathcal{V}(s)$,
- $M, s \models_k \neg p$ *iff* $p \notin \mathcal{V}(s)$, • $M, s \models_k \alpha \vee \beta$ *iff* $M, s \models_k \alpha$ or $M, s \models_k \beta$,
- $M, s \models_k \alpha \wedge \beta$ *iff* $M, s \models_k \alpha$ and $M, s \models_k \beta$,
- $M, s \models_k \mathrm{EX}\alpha$ *iff* $k > 0$ *and* $(\exists \pi \in \Pi_k(s))M, \pi(1) \models_k \alpha$,
- $M, s \models_k \mathrm{E}(\alpha \mathrm{U}_I \beta)$ *iff* $(\exists \pi \in \Pi_k(s))(\exists 0 \leq m \leq k)(m \in I$ *and* $M, \pi(m) \models_k \beta$ *and* $(\forall 0 \leq j < m) M, \pi(j) \models_k \alpha)$,
- $M, s \models_k \mathrm{EG}_I \alpha$ *iff* $(\exists \pi \in \Pi_k(s))((k \geq right(I)$ *and* $(\forall j \in I)\ M, \pi(j) \models_k \alpha)$ *or* $(k < right(I)$ *and* $(\exists l \in loop(\pi))(\forall min(left(I), l) \leq j < k) M, \pi(j) \models_k \alpha))$,
- $M, s \models_k \overline{\mathrm{K}}_c \alpha$ *iff* $(\exists \pi \in \Pi_k(\iota))(\exists 0 \leq j \leq k)(M, \pi(j) \models_k \alpha$ *and* $s \sim_c \pi(j))$,
- $M, s \models_k \overline{Y}_\Gamma \alpha$ *iff* $(\exists \pi \in \Pi_k(\iota))(\exists 0 \leq j \leq k)(M, \pi(j) \models_k \alpha$ *and* $s \sim_\Gamma^Y \pi(j))$, where $Y \in \{\mathrm{D}, \mathrm{E}, \mathrm{C}\}$,
- $M, s \models_k \overline{\mathcal{O}}_c \alpha$ *iff* $(\exists \pi \in \Pi_k(\iota))(\exists 0 \leq j \leq k)(M, \pi(j) \models_k \alpha$ *and* $s \bowtie_c \pi(j))$,
- $M, s \models_k \underline{\widehat{\mathrm{K}}}_c^d \alpha$ *iff* $(\exists \pi \in \Pi_k(\iota))(\exists 0 \leq j \leq k)(M, \pi(j) \models_k \alpha$ *and* $s \sim_c \pi(j)$ *and* $s \bowtie_d \pi(j))$.

An RTECTLKD formula φ is *valid in model M with bound k* (denoted $M \models_k \varphi$) iff $M, \iota \models_k \varphi$, i.e., φ is k-true at the initial state of the model M. The *bounded model checking problem* asks whether there exists $k \in \mathbb{N}$ such that $M \models_k \varphi$.

The following theorem states that for a given model and formula there exists a bound k such that the model checking problem ($M \models \varphi$) can be reduced to the bounded model checking problem ($M \models_k \varphi$). Its proof can be done by straightforward induction on the length of an RTECTLKD formula.

Theorem 1. *Let M be a model and φ an* RTECTLKD *formula. Then, the following equivalence holds: $M \models \varphi$ iff there exists $k \geq 0$ such that $M \models_k \varphi$.*

Further, by straightforward induction on the length of an RTECTLKD formula φ, we can show that φ is k-true in M if and only if φ is k-true in M with a number of k-paths reduced to $f_k(\varphi)$, where the function f_k : RTECTLKD $\rightarrow \mathbb{N}$ is defined as follows. $f_k(\mathbf{true}) = f_k(\mathbf{false}) = f_k(p) = f_k(\neg p) = 0$, where $p \in \mathcal{PV}$; $f_k(\alpha \wedge \beta) = f_k(\alpha) + f_k(\beta)$; $f_k(\alpha \vee \beta) = max\{f_k(\alpha), f_k(\beta)\}$; $f_k(\mathrm{E}(\alpha \mathrm{U}_I \beta)) = k \cdot f_k(\alpha) + f_k(\beta) + 1$; $f_k(\mathrm{EG}_I \alpha) = (k+1) \cdot f_k(\alpha) + 1$; $f_k(\overline{\mathrm{C}}_\Gamma \alpha) = f_k(\alpha) + k$; $f_k(Y\alpha) = f_k(\alpha) + 1$ for $Y \in \{\mathrm{EX}, \overline{\mathrm{K}}_c, \overline{\mathcal{O}}_c, \underline{\widehat{\mathrm{K}}}_c^d, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{E}}_\Gamma\}$.

3 SAT-Based BMC for RTECTLKD

Let $M = (\iota, S, T, \{\sim_c\}_{c \in Ag}, \{\bowtie_c\}_{c \in Ag}, \mathcal{V})$ be a model, φ an RTECTLKD formula, and $k \geq 0$ a bound. The proposed BMC method is based on the BMC encoding presented in [16], and it consists in translating the problem of checking whether $M \models_k \varphi$ holds, to the problem of checking the satisfiability of the propositional formula $[M, \varphi]_k := [M^{\varphi, \iota}]_k \wedge [\varphi]_{M,k}$. The formula $[M^{\varphi, \iota}]_k$ encodes sets of k-paths of M, whose size are equal to $f_k(\varphi)$, and in which at least one path starts at the initial state of the model M. The formula $[\varphi]_{M,k}$ encodes a number of constraints that must be satisfied on these sets of k-paths for φ to be satisfied. Once this translation is defined, checking satisfiability of an RTECTLKD formula can be done by means of a SAT-solver.

In order to define the formula $[M, \varphi]_k$ we proceed as follows. We begin with an encoding of states of the given model M. Since the set of states of M is finite, each state

s of M can be encoded by a bit-vector, whose length r depends on the number of agents' local states. Thus, each state s of M can be represented by a vector $w = (w_1, \ldots, w_r)$ (called a *symbolic state*) of propositional variables (called *state variables*). A finite sequence $(w_0, \ldots, w_k)$ of symbolic states of length k is called a *symbolic k-path*. Since in general we may need to consider more than one symbolic k-path, we introduce a notion of the j-th symbolic k-path $(w_{0,j}, \ldots, w_{k,j})$, where $w_{i,j}$ are symbolic states for $0 \leq j < f_k(\varphi)$ and $0 \leq i \leq k$. Note that the exact number of symbolic k-paths depends on the checked formula φ, and it can be calculated by means of the function f_k.

Let $\sigma : SV \rightarrow \{0, 1\}$ be a *valuation of state variables* (a *valuation* for short). Each valuation induces the function $\boldsymbol{\sigma} : SV^r \rightarrow \{0, 1\}^r$ defined in the following way: $\boldsymbol{\sigma}((w_1, \ldots, w_r)) = (\sigma(w_1), \ldots, \sigma(w_r))$. Moreover, let SV denote the set of all the state variables, and $SV(w)$ denote the set of all the state variables occurring in a symbolic state w. Next, let w and w' be two symbolic states such that $SV(w) \cap SV(w') = \emptyset$. We define the following auxiliary propositional formulae:

- $I_s(w)$ is a formula over $SV(w)$ that is true for a valuation σ iff $\boldsymbol{\sigma}(w) = s$.
- $p(w)$ is a formula over w that is true for a valuation σ iff $p \in \mathcal{V}(\boldsymbol{\sigma}(w))$ (encodes a set of states of M in which $p \in \mathcal{PV}$ holds).
- $H(w, w')$ is a formula over $SV(w) \cup SV(w')$ that is true for a valuation σ iff $\boldsymbol{\sigma}(w) = \boldsymbol{\sigma}(w')$ (encodes equivalence of two global states).
- $H_c(w, w')$ is a formula over $SV(w) \cup SV(w')$ that is true for a valuation σ iff $l_c(\boldsymbol{\sigma}(w)) = l_c(\boldsymbol{\sigma}(w))$ (encodes equivalence of local states of agent c).
- $HO_c(w, w')$ is a formula over $SV(w) \cup SV(w')$ that is true for a valuation σ iff $l_c(\boldsymbol{\sigma}(w')) \in \mathcal{G}_c$ (encodes an accessibility of a global state in which agent c is functioning correctly).
- $\widehat{H}_c^d(w, w') := H_c(w, w') \wedge HO_d(w, w')$.
- $\mathcal{R}(w, w')$ is a formula over $SV(w) \cup SV(w')$ that is true for a valuation σ iff $(\boldsymbol{\sigma}(w), \boldsymbol{\sigma}(w')) \in T$ (encodes the transition relation of M).
- Let $j \in \mathbb{N}$, and I be an interval. Then $In(j, I) :=$ **true** if $j \in I$, and $In(j, I) :=$ **false** if $j \notin I$.

Let $W = \{SV(w_{i,j}) \mid 0 \leq i \leq k \text{ and } 0 \leq j < f_k(\varphi)\}$ be a set of state variables. The propositional formula $[M^{\varphi,\iota}]_k$ is defined over the set W in the following way:

$$[M^{\varphi,\iota}]_k := I_\iota(w_{0,0}) \wedge \bigwedge_{j=0}^{f_k(\varphi)-1} \bigwedge_{i=0}^{k-1} \mathcal{R}(w_{i,j}, w_{i+1,j})$$

The next step of the reduction to SAT is the transformation of an RTECTLKD formula φ into a propositional formula $[\varphi]_{M,k} := [\varphi]_k^{[0,0,F_k(\varphi)]}$, where $F_k(\varphi) = \{j \in \mathbb{N} \mid 0 \leq j < f_k(\varphi)\}$, and $[\varphi]_k^{[m,n,A]}$ denotes the translation of φ at the symbolic state $w_{m,n}$ using k-paths, whose indices are in the set A.

Following [17], to translate an RTECTLKD formula with an operator Q (where $\mathrm{Q} \in \{\mathrm{EX}, \mathrm{EU}_I, \mathrm{EG}_I, \overline{\mathrm{K}}_1, \ldots, \overline{\mathrm{K}}_n, \overline{\mathcal{O}}_1, \ldots, \overline{\mathcal{O}}_n, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{E}}_\Gamma\} \cup \{\underline{\widehat{\mathrm{K}}}_c^d \mid c, d \in Ag \text{ and } c \neq d\}$), we want exactly one path to be chosen for translating the operator Q, and the remaining k-paths to be used to translate arguments of Q. To accomplish this goal we need some auxiliary functions. However, before we define them, we first recall a definition of a relation $\prec$ that is defined on the power set of $\mathbb{N}$ as follows: $A \prec B$ iff for all natural numbers x and y, if $x \in A$ and $y \in B$, then $x < y$. Notice that from the definition of $\prec$ it follows that $A \prec B$ iff either $A = \emptyset$ or $B = \emptyset$ or $A \neq \emptyset$, $B \neq \emptyset$, $A \cap B = \emptyset$ and $max(A) < min(B)$.

Now, let $A \subset \mathbb{N}$ be a finite nonempty set, $k, p \in \mathbb{N}$, and $m \in \mathbb{N}$ such that $m \leq |A|$:

- $g_l(A, m)$ denotes the subset B of A such that $|B| = m$ and $B \prec A \setminus B$.
- $g_r(A, m)$ denotes the subset C of A such that $|C| = m$ and $A \setminus C \prec C$.
- $g_s(A)$ denotes the set $A \setminus \{min(A)\}$.
- If $k+1$ divides $|A|-1$, then $h_{\mathrm{G}}(A, k)$ denotes the sequence $(B_0, \ldots, B_k)$ of subsets of $A \setminus \{min(A)\}$ such that $\bigcup_{j=0}^{k} B_j = A \setminus \{min(A)\}$, $|B_0| = \ldots = |B_k|$, and $B_i \prec B_j$ for every $0 \leq i < j \leq k$. If $h_{\mathrm{G}}(A, k) = (B_0, \ldots, B_k)$, then $h_{\mathrm{G}}(A, k)(j)$ denotes the set B_j, for every $0 \leq j \leq k$.
  Notice that if $k + 1$ does not divide $|A| - 1$, then $h_{\mathrm{G}}(A, k)$ is undefined. However, for every set A such that $|A| = f_k(\mathrm{EG}_I \alpha)$, it follows from the definition of f_k that $k + 1$ divides $|A| - 1$.
- If k divides $|A| - 1 - p$, then $h_{\mathrm{U}}(A, k, p)$ denotes the sequence $(B_0, \ldots, B_k)$ of subsets of $A \setminus \{min(A)\}$ such that $\bigcup_{j=0}^{k} B_j = A \setminus \{min(A)\}$, $|B_0| = \ldots = |B_{k-1}|$, $|B_k| = p$, and $B_i \prec B_j$ for every $0 \leq i < j \leq k$. If $h_{\mathrm{U}}(A, k, p) = (B_0, \ldots, B_k)$, then $h_{\mathrm{U}}(A, k, p)(j)$ denotes the set B_j, for every $0 \leq j \leq k$.
  Notice that if k does not divide $|A|-1-p$, then $h_{\mathrm{U}}(A, k, p)$ is undefined. However, for every set A such that $|A| = f_k(\mathrm{E}(\alpha \mathrm{U}_I \beta))$, it follows from the definition of f_k that k divides $|A| - 1 - f_k(\beta)$.

Let φ be an RTECTLKD formula, and $k \geq 0$ a bound. We define inductively the translation of φ over path number $n \in F_k(\varphi)$ starting at symbolic state $w_{m,n}$ as shown below. Let $min(A) = A'$, then:

- $[\mathbf{true}]_k^{[m,n,A]} := \mathbf{true}$, • $[\mathbf{false}]_k^{[m,n,A]} := \mathbf{false}$,
- $[p]_k^{[m,n,A]} := p(w_{m,n})$, • $[\neg p]_k^{[m,n,A]} := \neg p(w_{m,n})$,
- $[\alpha \wedge \beta]_k^{[m,n,A]} := [\alpha]_k^{[m,n,g_l(A,f_k(\alpha))]} \wedge [\beta]_k^{[m,n,g_r(A,f_k(\beta))]}$,
- $[\alpha \vee \beta]_k^{[m,n,A]} := [\alpha]_k^{[m,n,g_l(A,f_k(\alpha))]} \vee [\beta]_k^{[m,n,g_l(A,f_k(\beta))]}$,
- $[\mathrm{EX}\alpha]_k^{[m,n,A]} := \begin{array}{ll} (1)\ H(w_{m,n}, w_{0,A'}) \ \wedge\ [\alpha]_k^{[1,A',g_s(A)]}, & \text{if } k > 0 \\ (2)\ \mathbf{false}, & \text{otherwise} \end{array}$
- $[\mathrm{E}(\alpha \mathrm{U}_I \beta)]_k^{[m,n,\,A]} := H(w_{m,n}, w_{0,A'}) \wedge \bigvee_{i=0}^{k}([\beta]_k^{[i,A',h_{\mathrm{U}}(A,k,f_k(\beta))(k)]}$
  $\wedge In(i, I) \wedge \bigwedge_{j=0}^{i-1}[\alpha]_k^{[j,\,A',h_{\mathrm{U}}(A,k,f_k(\beta))(j)]})$,
- $[\mathrm{EG}_I\alpha]_k^{[m,n,A]} := H(w_{m,n}, w_{0,A'}) \wedge$
  (1) $\bigwedge_{j=left(I)}^{right(I)}[\alpha]_k^{[j,A',h_{\mathrm{G}}(A,k)(j)]}$, if $right(I) \leq k$
  (2) $\bigvee_{l=0}^{k-1}(H(w_{k,A'}, w_{l,A'}) \wedge \bigwedge_{j=min(left(I),l)}^{k-1}[\alpha]_k^{[j,A',h_{\mathrm{G}}(A,k)(j)]})$, otherwise.
- $[\overline{\mathrm{K}}_c\alpha]_k^{[m,n,A]} := I_\iota(w_{0,A'}) \wedge \bigvee_{j=0}^{k}([\alpha]_k^{[j,A',g_s(A)]} \wedge H_c(w_{m,n}, w_{j,A'}))$,
- $[\overline{\mathcal{O}}_c\alpha]_k^{[m,n,A]} := I_\iota(w_{0,A'}) \wedge \bigvee_{j=0}^{k}([\alpha]_k^{[j,A',g_s(A)]} \wedge HO_c(w_{m,n}, w_{j,A'}))$,
- $[\underline{\widehat{\mathrm{K}}}_c^d\alpha]_k^{[m,n,A]} := I_\iota(w_{0,A'}) \wedge \bigvee_{j=0}^{k}([\alpha]_k^{[j,A',g_s(A)]} \wedge \widehat{H}_c^d(w_{m,n}, w_{j,A'}))$,
- $[\overline{\mathrm{D}}_\Gamma\alpha]_k^{[m,n,A]} := I_\iota(w_{0,A'}) \wedge \bigvee_{j=0}^{k}([\alpha]_k^{[j,A',g_s(A)]} \wedge \bigwedge_{c\in\Gamma} H_c(w_{m,n}, w_{j,A'}))$,
- $[\overline{\mathrm{E}}_\Gamma\alpha]_k^{[m,n,A]} := I_\iota(w_{0,A'}) \wedge \bigvee_{j=0}^{k}([\alpha]_k^{[j,A',g_s(A)]} \wedge \bigvee_{c\in\Gamma} H_c(w_{m,n}, w_{j,A'}))$,
- $[\overline{\mathrm{C}}_\Gamma\alpha]_k^{[m,n,A]} := [\bigvee_{j=1}^{k}(\overline{\mathrm{E}}_\Gamma)^j\alpha]_k^{[m,n,A]}$.

The theorem below states the correctness and the completeness of the presented translation. It can be proven by induction on the complexity of the given RTECTLKD formula.

Theorem 2. *Let M be a model, and φ an* RTECTLKD *formula. Then for every $k \in \mathbb{N}$, $M \models_k \varphi$ if, and only if, the propositional formula $[M, \varphi]_k$ is satisfiable.*

Now, from Theorems 1 and 2 we get the following.

Corollary 1. *Let M be a model, and φ an* RTECTLKD *formula. Then, $M \models \varphi$ if, and only if, there exists $k \in \mathbb{N}$ such that the propositional formula $[M, \varphi]_k$ is satisfiable.*

4 Experimental Results

Our implementation of the presented BMC method uses Reduced Boolean Circuits (RBC) [1] to represent the propositional formula $[M, \varphi]_k$. An RBC represents subformulae of $[M, \varphi]_k$ by fresh propositions such that each two identical subformulae correspond to the same proposition. Further, our SAT-BMC method for RTCTLKD is, to our best knowledge, the first ones formally presented in the literature. However, to assess how well the new BMC algorithm performs, we compare it with non-BMC BDD-based symbolic model checking algorithm for ECTLKD that is implemented in McMAS (http://www-lai.doc.ic.ac.uk/mcmas/).

The tests have been performed on a computer with Intel Xeon 2 GHz processor and 4 GB of RAM, running Linux 2.6, with the default limits of 2 GB of memory and 5400 seconds. The specifications for the described benchmark are given in the universal form, for which we verify the corresponding counterexample formula, i.e., the formula which is negated and interpreted existentially.

FTC. To evaluate our BMC techniques, we analyse a scalable multi-agent system, which is a faulty train controller system (FTC). Figure 1 presents a DIIS composed of three agents: a controller and two trains, but in general the system consists of a controller, and n trains (for $n \geq 2$) that use their own circular tracks for travelling in one direction (states Away (A)). At one point, all trains have to pass through a tunnel (states Tunnel 'T'), but because there is only one track in the tunnel, trains arriving from each direction cannot use it simultaneously. There are colour light signals on both sides of the tunnel, which can be either red (state 'R') or green (state 'G'). All trains notify the controller when they request entry to the tunnel or when they leave the tunnel. The controller controls the colour of the colour light signals, however it can be faulty (state 'F'), and thereby it does not serve its purpose. In the figure, the initial states of the controller and the trains are 'G' and 'W' (Waiting in front of the tunnel) respectively, and the transitions with the same label are synchronised. Null actions are omitted in the figure.

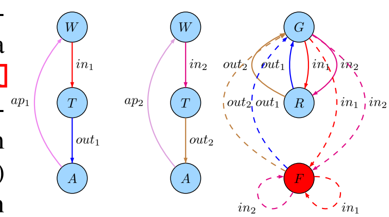


Fig. 1. A DIIS of FTC for 2 trains

Let $\mathcal{PV} = \{inTunnel_1, \ldots inTunnel_n, Red\}$ be a set of propositional variables, which we find useful in analysis of the scenario of the FTC system. A valuation function $\mathcal{V} : S \to 2^{\mathcal{PV}}$ is defined as follows. Let $Ag = \{Train1\ (T1), \ldots, TrainN\ (TN), Controller\ (C)\}$. Then, $inTunnel_c \in \mathcal{V}(s)$ if $l_c(s) = T$ and $c \in Ag \setminus \{C\}$; $Red \in \mathcal{V}(s)$ if $l_C(s) = R$. The specifications are the following:

φ_1 = $\mathrm{AG}_{[0,\infty]}\, \mathcal{O}_C(\bigwedge_{i=1}^{n-1} \bigwedge_{j=i+1}^{n} \neg(InTunnel_i \wedge InTunnel_j))$. "Always when *Controller* is functioning correctly, trains have exclusive access to the tunnel".

φ_2 = $\mathrm{AG}_{[0,\infty]}\mathcal{O}_C(inTunnel_1 \Rightarrow \mathrm{K}_{T1}(\neg inTunnel_2))$. "Always when *Controller* is functioning correctly, then if *Train1* is in the tunnel, it knows that *Train2* is not in the tunnel".

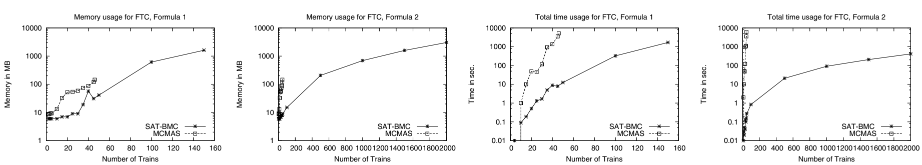
φ_3 = $\mathrm{AG}_{[0,\infty]}(inTunnel_1 \Rightarrow \widehat{\mathrm{K}}^C_{T1}(\bigwedge_{i=2}^{n}(\neg inTunnel_i)))$. "Always when *Train1* is in the tunnel, it knows under assumption that *Controller* is functioning correctly that none of the other trains is in the tunnel".

φ_4 = $\mathrm{AG}_{[0,\infty]}(inTunnel_1 \Rightarrow \widehat{\mathrm{K}}^C_{T1}(Red))$. "Always when *Train1* is in the tunnel, it knows under assumption that *Controller* is functioning correctly that the colour of the light signal for other trains is red".

φ_5 = $\mathrm{AG}_{[0,\infty]}(InTunnel_1 \Rightarrow \mathrm{K}_{T1}(\mathrm{AF}_{[1,n+1]}(\bigvee_{i=1}^{n} InTunnel_i)))$. "Always when *Train1* is in the tunnel, it knows that either he or other train will be in the tunnel during the next $n+1$ time units".

All the above properties are false in our DIIS model of the FTC system.

Performance Evaluation. The experimental results show that our SAT-based BMC significantly outperforms the BDD-based unbounded algorithm of McMAS for φ_1 and φ_2 in both the memory consumption and the execution time (as shown below in the line plots); note that both formulae are in ECTLKD. In the case of φ_1 our SAT-BMC is 3-times better than McMAS, and for φ_2 it is even 43-times better. A noticeable superiority of SAT-BMC for φ_1 and φ_2 follows from the long encoding times of the BDD for the transition relation and very short counterexamples.



Since McMAS does not support the $\widehat{\mathrm{K}}$ modality, we were not able to compare our results with McMAS for the formulae φ_3 and φ_4. Thus, we present results of our method only. Namely, for φ_3 and φ_4 we managed to compute the results for 1100 and 3000 trains, respectively, in the time of 5400 seconds (exact data for 1100 trains: $k = 4$, $f_k(\varphi_3) = 2$, encoding time (bmcT) is 210.12, memory use for encoding (bmcM) is 655.20, satisfiability checking time (satT) is 5258.43, memory use for satisfiability checking (satM) is 1412.00, bmcT+satT is 5468.55, max(bmcM,satM) is 1412.00; exact data for 3000 trains: $k = 1$, $f_k(\varphi_4) = 2$, bmcT is 170.38, bmcM is 1191.00, satT is 18.13, satM is 2356.00, bmcT+satT is 188.51, max(bmcM,satM) is 2356.00).

The formula φ_5 demonstrate that SAT-BMC is indeed a complementary technique to BDD-based unbounded model checking. McMAS was able to check φ_5 (in its equivalent ECTLKD form) for 45 trains in the time of 5400 seconds (memory use: 120MB), and our SAT-BMC succeed to compute the results only for 11 trains (exact data for 11 trains: $k = 21$, $f_k(\varphi_4) = 3$, bmcT is 1.99, bmcM is 4.47, satT is 4914.08, satM is 224.00, bmcT+satT is 4916.07, max(bmcM,satM) is 224.00). The reason for this is that the length of the counterexamples grows with the number of trains, i.e, for n trains $k = 2n - 1$.

Our future work will involve an implementation of the method also for other models of multi-agent systems, for example for standard interpreted systems. Moreover, we are going to define a BDD-based BMC algorithm for RTECTLKD, and compare it with the method presented in this paper.

References

1. Abdulla, P.A., Bjesse, P., Eén, N.: Symbolic Reachability Analysis Based on SAT-Solvers. In: Graf, S. (ed.) TACAS 2000. LNCS, vol. 1785, pp. 411–425. Springer, Heidelberg (2000)
2. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. The MIT Press, Cambridge (1999)
3. Sistla, A.P., Emerson, E.A., Mok, A.K., Srinivasan, J.: Quantitative temporal reasoning. Real-Time Systems 4(4), 331–352 (1992)
4. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning about Knowledge. MIT Press, Cambridge (1995)
5. Jones, A., Lomuscio, A.: A BDD-based BMC approach for the verification of multi-agent systems. In: CS&P 2009, vol. 1, pp. 253–264. Warsaw University (2009)
6. Kacprzak, M., Lomuscio, A., Lasica, T., Penczek, W., Szreter, M.: Verifying Multi-agent Systems via Unbounded Model Checking. In: Hinchey, M.G., Rash, J.L., Truszkowski, W.F., Rouff, C.A. (eds.) FAABS 2004. LNCS (LNAI), vol. 3228, pp. 189–212. Springer, Heidelberg (2004)
7. Lomuscio, A., Penczek, W., Qu, H.: Partial order reduction for model checking interleaved multi-agent systems. In: AAMAS 2010, pp. 659–666. IFAAMAS Press (2010)
8. Lomuscio, A., Sergot, M.: Deontic interpreted systems. Studia Logica 75(1), 63–92 (2003)
9. Męski, A., Penczek, W., Szreter, M.: Bounded model checking linear time and knowledge using decision diagrams. In: CS&P 2011, pp. 363–375. Biaystok University of Technology (2011)
10. Penczek, W., Lomuscio, A.: Verifying epistemic properties of multi-agent systems via bounded model checking. In: AAMAS 2003, pp. 209–216. ACM (2003)
11. Penczek, W., Woźna-Szcześniak, B., Zbrzezny, A.: Towards SAT-based BMC for LTLK over interleaved interpreted systems. In: CS&P 2011, pp. 565–576. Białystok University of Technology (2011)
12. Raimondi, F., Lomuscio, A.: Symbolic Model Checking of Deontic Interpreted Systems via OBDDs. In: Lomuscio, A., Nute, D. (eds.) DEON 2004. LNCS (LNAI), vol. 3065, pp. 228–242. Springer, Heidelberg (2004)
13. Raimondi, F., Lomuscio, A.: Automatic verification of multi-agent systems by model checking via OBDDs. Journal of Applied Logic 5(2), 235–251 (2005)
14. Wooldridge, M.: An introduction to multi-agent systems. John Wiley, England (2002)
15. Woźna, B., Lomuscio, A., Penczek, W.: Bounded model checking for deontic interpreted systems. In: LCMAS 2004. ENTCS, vol. 126, pp. 93–114. Elsevier (2005)
16. Woźna-Szcześniak, B., Zbrzezny, A., Zbrzezny, A.: The BMC Method for the Existential Part of RTCTLK and Interleaved Interpreted Systems. In: Antunes, L., Pinto, H.S. (eds.) EPIA 2011. LNCS (LNAI), vol. 7026, pp. 551–565. Springer, Heidelberg (2011)
17. Zbrzezny, A.: Improving the translation from ECTL to SAT. Fundamenta Informaticae 85(1-4), 513–531 (2008)

A Concept for Testing Robustness and Safety of the Context-Aware Behaviour of Autonomous Systems

Zoltán Micskei, Zoltán Szatmári, János Oláh, and István Majzik

Budapest University of Technology and Economics, Budapest, Hungary
{micskeiz,szatmari,olahj,majzik}@mit.bme.hu

Abstract. Autonomous systems are used nowadays in more and more sectors from vehicles to domestic robots. They can make decisions on their own or interact with humans, thus their robustness and safety are properties of crucial importance. Due to the adaptive and context-aware nature of these systems, the testing of such properties is especially challenging. In this paper, we propose a model-based testing approach to capture the context and requirements of such systems, to automatically generate test data representing complex situations, and to evaluate test traces and compute test coverage metrics.

Keywords: testing, autonomous systems, context modelling, test coverage.

1 Introduction

An autonomous system (AS) can be defined as one that makes and executes decisions to achieve a goal without full, direct human control [1]. Notable characteristics shared by the different kinds of autonomous systems include reasoning, learning, adaptation and context-awareness. In many scenarios, several autonomous systems are working together to reach a common goal, thus communication and cooperation are also important, and the interacting partners form part of the context of an individual AS.

A typical example of an AS is an autonomous robot, which is working in a real, uncontrolled environment, possibly in the presence of humans. Even if the task of a robot is relatively simple, e.g., to pick up garbage from the ground, it should be able to differentiate and recognize numerous types of objects and be prepared to take into account the unexpected movements of humans. Thus, it shall be *robust* in order to be capable of handling unforeseen situations and *safe* to avoid harmful effects with respect to humans. The prerequisite of the application of autonomous systems is a thorough verification of these requirements.

In the R3-COP project[1] our work is focused on testing the context-aware behaviour of autonomous robots, especially the robustness and functional safety of their behaviour. Precisely speaking, robustness is an attribute of dependability, which measures the degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions. Thus, the goal of

[1] Resilient Reasoning Robotic Co-operating Systems, `http://www.r3-cop.eu/`

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 504–513, 2012.

robustness testing is the generation of these situations to detect the potential design faults that result in incorrect operation. Functional safety is defined as freedom from unacceptable occurrence of harm (characterized with a given probability and severity), which depends on the correct functioning of the system. Testing functional safety is performed by executing the system in a controlled way to demonstrate the absence of unsafe behaviour. In our work we restrict the focus of testing robustness and functional safety to the design faults affecting context-aware behaviour and do not deal with testing the effects of random operational faults.

Testing the robustness and safety in autonomous systems is especially challenging and it requires the development of new methods due to the following characteristics. First, the behaviour is *highly context-aware*: the actual behaviour of an AS depends not only on the events it receives, but also on the perceived state of the environment (that is typically stored as a context model in the AS). Second, the context is complex and there are a *large number of possible situations*: in real physical world the number and types of potential context objects, attributes and interactions that need to be specified can be large. Third, *adaptation to evolving context* is required: as most of the autonomous systems contain some kind of learning and reasoning capabilities, their behaviour can change in time based on feedback from the evolving environment.

These characteristics have also consequences on the specification of the requirements to be tested. Full behaviour specification can be impractical due to the complexity of the behaviour and the diversity of the system environments, and requirements should include the evolution of the environment. It means that the developed test cases should not only include the input signals and messages to the system under test (SUT), but they should contain a sufficiently detailed description about the environment of the system.

As stated in a recent paper [6], testing autonomous systems is still an unsolved key area. In the project, considering our goals, we identified the following insufficiencies of existing testing approaches and proposed the following solutions:

- *Lack of easy-to-use mechanisms to express and formalize context-aware behaviour* (although these mechanisms are a prerequisite of requirements-based automated test generation and test evaluation). Most noticeably, in existing standard test description languages there is no support to express changes in the context [2]. To overcome this problem of capturing test requirements, we defined a language which is based on *context models* and *scenario based behaviour specification*. The context model allows systematically capturing domain knowledge about the context of the system. Hierarchy in the modelling can efficiently support handling of the types of diverse objects in the environment. Context models may also represent the existence of cooperating partners and humans, as well as the explicit messages and commands from them. With respect to the dynamics of the system behaviour, the scenario based language provides a lightweight formalism (that is close to the engineers' way of thinking) to capture the behaviour (actions and messages) of a SUT in case of a test context. The application of modalities in the scenario language allows expressing prohibited behaviour (that would violate safety) and potentially allowed behaviour (this way capturing learning and adaptive behaviour).

- *Ad-hoc testing of stressful conditions and extreme situations*: Previous research focused first of all on producing high fidelity simulators [4] or executing excessive field testing [5] for the verification of AS. There exist methods for testing the physical aspects; however, not all behavioural aspects are well-covered [3]. Our proposed solution is based on context modelling: we included in the context models the constraints and conditions that determine the normal and exceptional situations and defined methods to *systematically generate* stressful test contexts by violating the constraints, reaching boundary conditions, and combining contexts from various requirements (to test the implementation in case of interleaving scenarios).
- *Lack of precise and objective test coverage metrics* that can characterize the thoroughness of the testing process. On the basis of context models we defined precise coverage metrics, especially *robustness related metrics* that refer to constraints and conditions (to be violated), and combinations of context elements and fragments. Different coverage criteria were used to define the test goals and search-based techniques were applied to generate the required complex test suite. The formalization of robustness and safety related requirements allows an automated evaluation of test executions (especially in a simulated environment) and thus deriving concrete success and coverage measures that characterize the testing process and help identifying weak points in testing.

The rest of the paper presents our model based testing framework that is being finalized and validated in the R3-COP project. Section 2 specifies the testing approach (the goals and the components of the framework). Section 3 presents how the test requirements are captured, i.e., the context and scenario modelling. Section 4 outlines the test generation approach, while Section 5 presents the test evaluation and the main categories of the applied coverage metrics. The detailed presentation of all technical details will be publicly available in an R3-COP project deliverable (D4.2.1).

2 Testing Approach

Our work focuses on testing the control software of an individual AS that determines how the AS reacts (i.e., what actions it sends to the actuators) to the events from the perception or the communication components representing the (changes in the) context of the AS. The control software is treated as a black-box and the testing is concentrated on the system-level behaviour: detecting misbehaviour with respect to safety and robustness properties in diverse situations. Although the scope of testing is the behaviour of an individual AS, cooperation among systems can also be represented in the requirements, since input messages and input events (with respect to the movements of other systems and humans) as well as output messages are part of the context model. However, the overall goal of a cooperative behaviour is not tested. As safety and robustness failures manifest when several typical situations combine in an unexpected manner (e.g., an autonomous vehicle receives a new direction just when it detects that a pedestrian crossing and another vehicle approaching from the next lane), our approach focuses on covering such problematic complex (combined) situations. Testing is carried out in an interactive simulator environment, where different test situations can be prepared and the system under test can react to the changes in the context as obtained through its sensors.

Fig. 1 gives a high-level overview of the testing process. Starting from the specification of the AS and the knowledge of the application domain, the process evaluates the safety and robustness of the control software and computes coverage metrics. The key components (that are novel in our approach) are the context and scenario modelling, the search-based test data generation on the basis of robustness related coverage criteria, and methods for automated test evaluation. Related new tools are the test data generator and the test oracle generator.

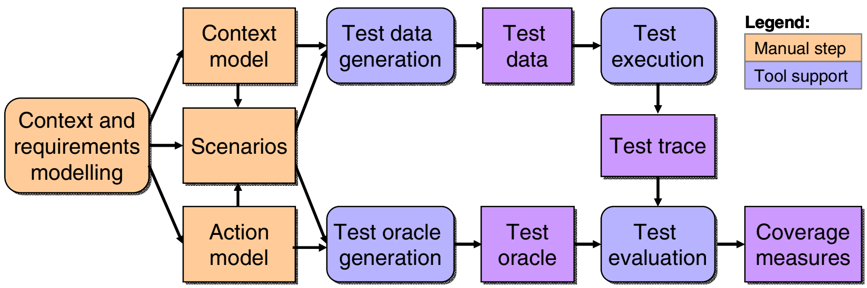


Fig. 1. Overview of the proposed approach

The rest of the paper walks through the steps of this testing process to show how the defined testing goals can be achieved by creating a context model, capturing requirements as scenarios, generating test data, executing the tests while recording test traces, and finally evaluating test traces with respect to the requirements.

To illustrate the developed testing approach a home *vacuum cleaner robot* is used as a running example. The robot is able to discover its surroundings when placed in a new room, create a map of its environment, and clean the room. The robot should avoid collision with living beings and should perform some limited surveillance tasks (e.g., detect unusual noises). Note that the purpose of the example is only to illustrate the testing concepts, and it is not intended to be a full system and test specification.

3 Formalizing Application Requirements

Usually application requirements are available as natural language specifications, as use case descriptions or as some text structured in tables. In order to use such requirements in test evaluation or in automated test data generation, a more structured format is needed. In general, to implement the testing, the formalization of at least two artefacts is required.

1. *Test data*. The input to the system under test should be specified. As described previously, in case of autonomous systems this should include (i) the initial state of the context of the system, (ii) its evolution in time, and (iii) the messages and

commands received by the SUT. These concepts are captured in a *context model*. Outputs of the SUT are included in a separate *action model*.

2. *Test oracle*. The responsibility of the test oracle is to evaluate the test outcome, the actions and output messages of the system. As specifying the exact outcome of every situation could be infeasible, a lightweight approach is used. The requirements are expressed as *scenarios* and are checked for every executed test (to detect potential safety and robustness failures).

3.1 Context Modelling

In order to have a structured way of describing the test data, first a *context model* is created. It consists of two parts. The *static part* represents the environment objects and their attributes in a type hierarchy (in the vacuum cleaner example it includes concepts like room, furniture inside a room, humans or animals). The *dynamic part* contains events as distinguished elements to represent changes with regard to objects (i.e., an object appears, disappears) and their relations and properties (e.g., a relation is formed or a property is transformed). The events have attributes and specific relations to the static objects depending on the type of the event.

Several modelling languages exist to express such models; see e.g. [7] for context modelling approaches. An easy to use approach to capture domain concepts is the use of *ontologies*. Existing ontologies related to robots (like KnowRob [8]) can be reused when defining the context model. To ease the programmatic manipulation of these context models (which is needed when test data is generated), ontology models are transformed systematically to *metamodels* (see e.g. Fig. 2 in case of the vacuum cleaner robot) and *instance models* conforming to this metamodel. The metamodels are extended with domain-specific constraints that for example require specific attribute values or define restrictions with respect to the number of objects in a model.

The structured context metamodel offers several advantages over an ad-hoc representation of context elements in the different requirements. It supports the automated generation of test data, including the completion and systematic combination

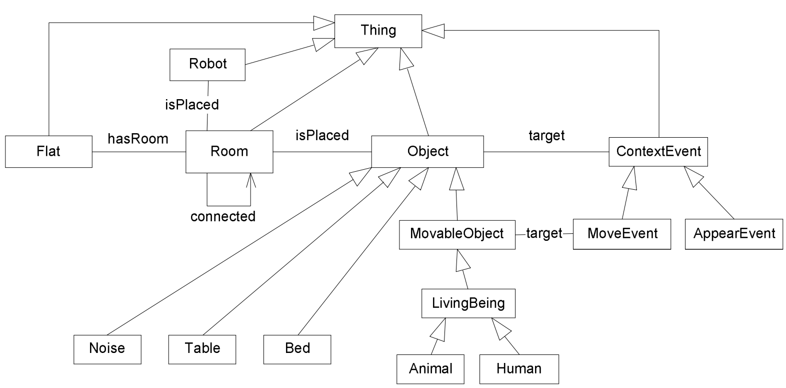


Fig. 2. Context metamodel for the vacuum cleaner robot

of context model elements from different requirements. The domain-specific constraints allow the automated generation of extreme contexts (as test data) that violate these constraints. Moreover, it is also the basis of the definition of precise test coverage metrics. In our framework we construct an OWL2 based domain ontology then map it automatically to metamodels to be manipulated in the Eclipse Modeling Framework[2].

3.2 Scenario Modelling

For test oracles, requirements are expressed as graphical scenarios in the form of extended UML 2 Sequence Diagrams that represent events/messages received and actions/messages sent by the SUT. Each diagram has two parts: (1) a trigger part (which means that the scenario is only relevant if its trigger part has been successfully traversed) that may have several fragments, like *opt* fragment for expressing optional behaviour, *alt* for expressing alternatives, *neg* for a negative fragment that should not happen and (2) an assert part that consist of an *assert* segment (that shall happen).

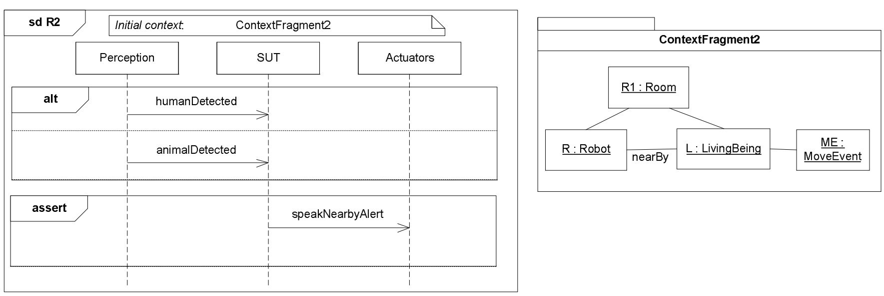


Fig. 3. Example scenario model R2: Alerting a living being

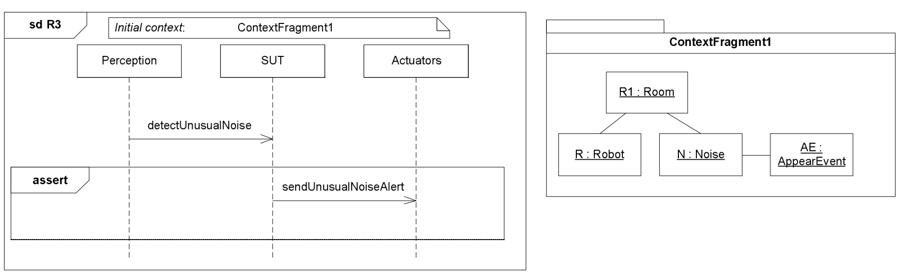


Fig. 4. Example scenario model R3: Detecting unusual noise

Language extensions were added to refer to (changing) contexts. Using these extensions, context model fragments are included as *initial context* of the scenario, *interim context* in the trigger part that should occur during test execution, and *final context* (in the *assert* part) that is checked to determine the success of the test.

[2] http://www.eclipse.org/modeling/emf/

Fig. 3 and Fig. 4 present two requirements (R2 and R3) of the vacuum cleaner. R2 states that when a living being is detected nearby the robot then it has to be alerted. R3 states that if a noise is detected in the room then the robot should send a predefined alert. Here the requirements with respect to the initial contexts are described using context fragments (model instances conforming to the context metamodel).

4 Generating Test Data

The context models provide a mechanism to describe environments, which could be later represented in a test execution setup (e.g., in a simulator). Requirements expressed as scenarios provide a way to check later these executions and search for misbehaviours. However, it is still missing how the different test data describing interesting environments and situations are acquired. By creating these test data by hand, some of the situations can be covered, but there is no guarantee that every stressful context is represented in the test data, or that the combinations of situations are completely checked.

We proposed an automated *test data generation* approach, which uses search-based techniques [9]. The key ingredients for the application of a search-based technique are the representation of the potential solutions and the definition of a fitness function. According to our approach, the solutions, i.e., the potential test data, will be generated as model instances conforming to the context metamodel. We use model transformation to manipulate the instances (adding/removing context elements and fragments to/from the candidate solutions) and a fitness function to guide the search. The fitness functions are based on the context requirements and on the test strategies (corresponding to test coverage metrics, see Section 5) that include the following:

- Creating the minimal number of context objects that are needed to test the satisfaction of a requirement. Furthermore, completing the minimal number of context objects with more and more additional objects from the context metamodel, potentially instantiating all types of the context metamodel in the set of test data.
- Combining context models related to different requirements to test whether the combined contexts will violate these requirements.
- Creating context models that are at the boundary or even violate systematically selected domain-specific constraints for robustness testing.

Fig. 5 presents the minimal test data generated by combining the initial context fragments belonging to requirements R2 and R3 as a stressful situation for the SUT. Although this example is simple, the generation of test data for a given test strategy is a non-trivial task due to the high number of context model elements, their type hierarchy and relations, and the constraints that have to be fulfilled/violated to get meaningful test data.

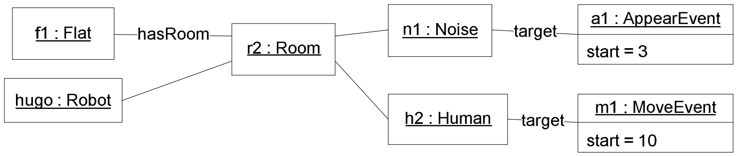


Fig. 5. Test data generated by combining the context fragments of requirements R2 and R3

A similar testing approach is presented in [10]: it utilizes so-called stakeholder soft-goals as quality functions and applies metaheuristic search-techniques to generate tests. However, there are essential differences in comparison with our framework. Their approach encodes the environment settings as a matrix of cells to represent object locations, while we propose the use of a high level metamodel to describe the possible contexts and use model instances conforming to this metamodel to represent concrete test data. We believe that this representation is more flexible and easy to use for domain experts. Another difference lies in the fitness calculation and execution of the tests. Their approach involves on-line testing: inserting the SUT into a test environment (made up of the generated test data) and calculating the fitness of potential test data based on the execution of the SUT. The test generation stops if the SUT violates the soft goal or there is no improvement observed after a number of generations and executions. On the contrary, we apply off-line test generation: we do not use the results of the execution of the SUT to assess the test data. In order to calculate the fitness of test data, we search for patterns that may trigger the formalized requirements and satisfy the test goals (coverage criteria). While our test data generation concept may not be adequate for the validation of every kind of soft goals, it does not depend on the domain and implementation of the SUT.

5 Evaluation of Test Results and Coverage

To be able to include the generated test data in a simulator, a lot of simulator-specific information should be hard-coded in the test data (e.g., how is the position of an object encoded). To avoid this dependency from simulators, test data creation consists of two steps. First, so called *abstract test data* are generated and then a post-processing step produces *concrete test data* in a format dependent on the simulator. This way the formalized requirements use only general, abstract concepts and relationships (e.g., the robot is *near to* something). These are replaced in the post-processing step with compatible types defined in the simulator and the exact parameters (e.g., physical coordinates) are assigned.

After test data are generated, the following steps have to be executed. The simulator is fed with each generated test data (which describe the environment of the SUT) and then it processes the dynamic part of the test data (i.e., the evolution of the context, sending and receiving of messages etc.). During test execution detailed test traces are captured that record the events and actions of the SUT with their timing and also changes in the context. Finally, each captured test trace is checked against each of the scenarios to identify whether a scenario is triggered (checking the trigger part) and violation of any requirement is detected (in the "assert" part of the scenario). In this way, the proper handling of interleaving requirements can also be verified.

The test evaluation consists of two different aspects. On one hand, the matching of events and actions has to be checked. On the other hand, the actual context has also to be matched to the context fragment specified in the requirement.

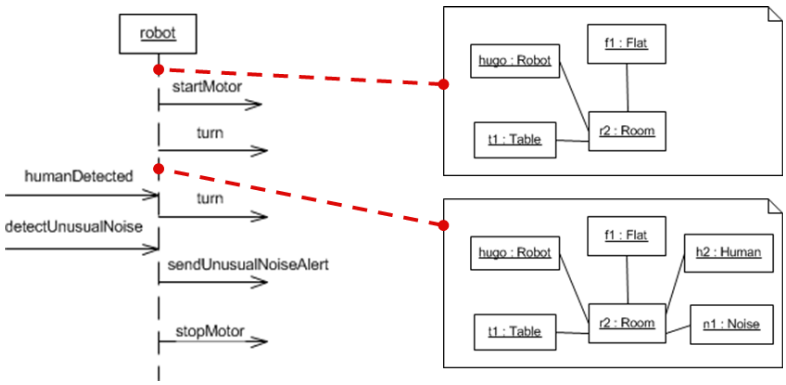


Fig. 6. Illustration of a test trace in case of testing the vacuum cleaner robot

Fig. 6 illustrates a test trace captured when the vacuum cleaner robot was tested using the test context presented in Fig. 5. According to this test data (the *AppearEvent* and *MoveEvent*) there was a change in the context: a human moved into the room and a noise appeared. When this test trace is evaluated with respect to requirements R2 and R3 (see in Fig. 3 and Fig. 4) the following can be derived: R3's initial context fragment can be matched with the second context, the message in its trigger part appears (there is a *detectUnusualNoise* message), and the action in its assert parts appears too, thus R3 is satisfied. In case of R2 the initial context fragment can be matched with the second context (when the distance between the position of the robot and the human satisfies the *nearBy* relation), the trigger part appears in the trace (there is a *humanDetected* message), but the action in the assert part does not appear in the trace. This way R2 is violated. This example demonstrated how the improper handling of stressful situations (i.e., interleaving of several potential scenarios) can be detected using a test context generated by combining the initial context fragments of different requirements. (Note that the matching of contexts and messages is obvious in this example due to the simplicity of the scenario, but in general it could be more complicated.)

Finally, when all the executed tests are evaluated, the quality of the testing should be assessed. One way to achieve this is to compute different coverage metrics. C*ontext related coverage metrics* measure what part of the context model has been covered during testing (e.g., whether there are objects, which have not been present in any test runs) and what combinations of initial context fragments from different requirements were covered. *Scenario related metrics* measure coverage on the scenarios, e.g., whether all scenarios have been triggered, or whether there were any violated requirements. *Robustness related metrics* measure the thoroughness of the generation of extreme contexts by considering the coverage of violated constraints and potential boundary values from the context model.

6 Summary and Future Work

This paper presented the challenges of testing autonomous systems and proposed a method to test the robustness and functional safety of the behaviour of the system's control. The proposed approach uses context modelling and graphical scenarios to capture the context and requirements of the system and automatically generates test data and test oracle to test complex or unforeseen situations. Once the test data is executed, the satisfaction of the requirements is checked and the coverage with respect to the context or scenarios is calculated. Currently we are working on the validation of the process using two real-world use cases, which involves refining the modelling notations, the test generation and the test evaluation algorithms.

Acknowledgements. This work was partially supported by the ARTEMIS JU and the Hungarian National Development Agency (NFÜ) in frame of the R3-COP project.

References

1. Connelly, J., Hong, W., Mahoney, R., Sparrow, D.: Challenges in Autonomous System Development. In: Proc. of Performance Metrics for Intelligent Systems (PerMIS 2006) Workshop (2006)
2. Waeselynck, H., et al.: Mobile Systems from a Validation Perspective: a Case Study. In: Proc. of 6th International Symposium on Parallel and Distributed Computing (ISPDC). IEEE Press (2007), doi:10.1109/ISPDC.2007.37
3. Michelson, R.C.: Test and Evaluation for Fully Autonomous Micro Air Vehicles. ITEA Journal 29(4), 367–374 (2008)
4. Scrapper, C., Balakirsky, S., Messina, E.: MOAST and USAR-Sim: A Combined Framework for the Development and Testing of Autonomous Systems. In: Proc. of SPIE 6230, 62301T (2006), doi:10.1117/12.663898
5. Kelly, A., et al.: Toward Reliable Off Road Autonomous Vehicles Operating in Challenging Environments. The International Journal of Robotics Research 25(5-6), 449–483 (2006), doi:10.1177/0278364906065543
6. Weiss, L.G.: Autonomous Robots in the Fog of War. IEEE Spectrum 48(8), 30–57 (2011), doi:10.1109/MSPEC.2011.5960163
7. Baldauf, M., Dustdar, S., Rosenberg, F.: A Survey on Context-aware Systems. Int. J. Ad Hoc Ubiquitous Comput. 2(4), 263–277 (2007), doi:10.1504/IJAHUC.2007.014070
8. Tenorth, M., Beetz, M.: KnowRob – Knowledge Processing for Autonomous Personal Robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4261–4266. IEEE Press, New York (2009), doi:10.1109/IROS.2009.5354602
9. Szatmári, Z., Oláh, J., Majzik, I.: Ontology-Based Test Data Generation Using Metaheuristics. In: Proc. 8th Int. Conf. on Informatics in Control, Automation, and Robotics (ICINCO), Noordwijkerhout, The Netherlands (2011)
10. Nguyen, C.D., Perini, A., Tonella, P., Miles, S., Harman, M., Luck, M.: Evolutionary Testing of Autonomous Software Agents. In: Proc. AAMAS (1), pp. 521–528 (2009)

Two Approaches to Bounded Model Checking for Linear Time Logic with Knowledge*

Artur Męski[1,2], Wojciech Penczek[2,3], Maciej Szreter[2],
Bożena Woźna-Szcześniak[4], and Andrzej Zbrzezny[4]

[1] Institute of Computer Science, PAS, Poland
{meski,penczek,mszreter}@ipipan.waw.pl
[2] University of Natural Sciences and Humanities, II, Poland
[3] University of Łódź, FMCS, Poland
[4] Jan Długosz University, IMCS, Poland
{b.wozna,a.zbrzezny}@ajd.czest.pl

Abstract. The paper deals with symbolic approaches to bounded model checking (BMC) for Linear Time Temporal Logic extended with the epistemic component (LTLK), interpreted over Interleaved Interpreted Systems. Two translations of BMC for LTLK to SAT and to operations on BDDs are presented. The translations have been implemented, tested, and compared with each other as well as with another tool on several benchmarks for MAS. Our experimental results reveal advantages and disadvantages of SAT- versus BDD-based BMC for LTLK.

1 Introduction

Verification of multi-agent systems (MAS) is an actively developing field of research. Several approaches based on model checking [5] have been put forward for the verification of MAS. Typically, they employ combinations of the epistemic logic with either branching [15] or linear time temporal logic [8,13,7]. Some approaches reduce the verification problem to the one for plain temporal logic [2,8], while others treat typical MAS modalities such as (distributed, common) knowledge as first-class citizens and introduce novel algorithms for them [13,15].

In an attempt to alleviate the state-space explosion problem (i.e., an exponential growth of the system state space with the number of the agents) two main approaches have been proposed based on combining bounded model (BMC) with symbolic verification using translations to either ordered binary decision diagrams (BDDs) [10] or propositional logic (SAT) [14]. However, the above approaches deal with the properties expressed in CTLK (i.e., CTL extended with an epistemic component) only.

In this paper we aim at completing the picture of applying BMC-based symbolic verification to MAS by looking at LTLK (i.e., LTL extended with an epistemic component, called also CKL_n [8]), interpreted on *interleaved interpreted systems* (IIS) [12]. IIS are a special class of interpreted systems [6] in which only one action at a time is performed in a global transition. Our original contribution consists in defining two novel model checking methods for LTLK, namely a SAT-based and a BDD-based BMC. The methods have been implemented, tested, and compared with each other as well as with

* Partly supported by National Science Center under the grant No. 2011/01/B/ST6/05317 and 2011/01/B/ST6/01477.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 514–523, 2012.

MCK [7] on three benchmarks for MAS. Our experimental results reveal advantages and disadvantages of LTLK SAT- versus BDD-based BMC for MAS, which are consistent with comparisons for temporal logics [3]. Although our methods are described for IIS, they can be applied to general interpreted systems [6] as well.

The rest of the paper is organised as follows. We begin in Section 2 by presenting IIS and the logic LTLK. In Section 3 we present SAT- and BDD-based BMC for LTLK. In the last section we discuss our experimental results and conclude the paper.

2 Interleaved Interpreted Systems and LTLK

The semantics of *interpreted systems* provides a setting to reason about MAS by means of specifications based on knowledge and linear or branching time. We report here the basic setting as popularised in [6], restricted to interleaved interpreted systems [12]. Therefore, we assume that if more than one agent is active at a given state, all the active agents perform the same (shared) action in the round. Note that it is still possible for agents to communicate by means of shared actions.

We begin by assuming a MAS to be composed of n agents $\mathcal{A}$; note we do not consider the environment component. This may be added with no technical difficulty at the price of heavier notation. We associate a set of *possible local states* L_q and *actions* Act_q to each agent $q \in \mathcal{A}$. We assume that the special action ϵ_q, called *null* action of agent q, belongs to Act_q; as it will be clear below the local state of agent q remains the same if the null action is performed. Also note that we do not assume that the sets of actions of the agents to be disjoint. We call $Act = \bigcup_{q \in \mathcal{A}} Act_q$ the union of all the sets Act_q. For each action a by $Agent(a) \subseteq \mathcal{A}$ we mean all the agents q such that $a \in Act_q$, i.e., the set of agents potentially able to perform a. Following closely the interpreted system model, we consider a *local protocol* modelling the program the agent is executing. Formally, for any agent q, the actions of the agents are selected according to a *local protocol* $P_q : L_q \to 2^{Act_q}$; we assume that $\epsilon_q \in P_q(l)$, for any $l \in L_q$, i.e., we insist on the null action to be enabled at every local state. For each agent q, we define an evolution (partial) function $t_q : L_q \times Act_q \to L_q$, where $t_q(l, \epsilon_q) = l$ for each $l \in L_q$. The local transition function considered here differs from the standard treatment in interpreted systems by having the local action as the parameter instead of the joint action. A *global state* $g = (l_1, \ldots, l_n)$ is a tuple of local states for all the agents in the MAS corresponding to an instantaneous snapshot of the system at a given time. Given a global state $g = (l_1, \ldots, l_n)$, we denote by $g^q = l_q$ the local component of agent $q \in \mathcal{A}$ in g. Let G be a set of global states. The *global interleaved evolution* function $t : G \times \prod_{q=1}^{n} Act_q \to G$ is defined as follows: $t(g, a_1, \ldots, a_n) = g'$ iff there exists an action $a \in Act \setminus \{\epsilon_1, \ldots, \epsilon_n\}$ such that for all $q \in Agent(a)$, $a_q = a$ and $t_q(g^q, a) = g'^q$, and for all $q \in \mathcal{A} \setminus Agent(a)$, $a_q = \epsilon_q$ and $t_q(g^q, \epsilon_q) = g^q$. In brief we write the above as $g \stackrel{a}{\longrightarrow} g'$. Similar to blocking synchronisation in automata, the above insists on all agents performing the same non-epsilon action in a global transition.

We assume that the global transition relation is total, i.e., that for any $g \in G$ there exists $a \in Act$ such that $g \stackrel{a}{\longrightarrow} g'$ for some $g' \in G$. An **infinite** sequence of global states and actions $\rho = g_0 a_0 g_1 a_1 g_2 \ldots$ is called an *interleaved path* (or simply a *path*) originating at g_0 if there is a sequence of interleaved transitions from g_0 onwards, i.e., $g_i \stackrel{a_i}{\longrightarrow} g_{i+1}$ for every $i \geq 0$. Any **finite** prefix of a path is called an *(interleaved) run*.

By $length(\rho)$ we mean the number of the states of ρ if ρ is a run, and ω if ρ is a path. In order to limit the indices range of ρ which can be a path or run, we define the relation $\trianglelefteq_\rho$. Let $\trianglelefteq_\rho \stackrel{def}{=} <$ if ρ is a path, and $\trianglelefteq_\rho \stackrel{def}{=} \leq$ if ρ is a run. The set of all the interleaved paths and runs originating from g is denoted by $\Pi(g)$. The set of all the interleaved paths originating from g is denoted by $\Pi^\omega(g)$. A state g is said to be *reachable* from g_0 if there is a path or a run $\rho = g_0 a_0 g_1 a_1 g_2 \ldots$ such that $g = g_i$ for some $i \geq 0$.

Let $\mathcal{PV}$ be a set of propositions. An *interleaved interpreted system (IIS)*, also referred to as a model, is a tuple $M = (G, \iota, \Pi, \{\sim_q\}_{q \in \mathcal{A}}, \mathcal{V})$, where G is a set of global states, $\iota \in G$ is an initial (global) state, $\Pi = \bigcup_{g \in G} \Pi(g)$ is the set of all the interleaved paths originating from all states in G, $\sim_q \subseteq G \times G$ is an *epistemic indistinguishability* relation for each agent $q \in \mathcal{A}$, defined by $g \sim_q r$ if $g^q = r^q$, and $\mathcal{V} : G \to 2^{\mathcal{PV}}$ is a valuation function.

LTLK. Let $\mathcal{PV}$ be a set of atomic propositions to be interpreted over the global states of a system, $p \in \mathcal{PV}$, and $\Gamma \subseteq \mathcal{A}$. Then, the syntax of LTLK is defined by the following BNF grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathrm{X}\varphi \mid \varphi\mathrm{U}\varphi \mid \varphi\mathrm{R}\varphi \mid$$
$$\mathrm{K}_q\varphi \mid \overline{\mathrm{K}}_q\varphi \mid \mathrm{E}_\Gamma\varphi \mid \overline{\mathrm{E}}_\Gamma\varphi \mid \mathrm{D}_\Gamma\varphi \mid \overline{\mathrm{D}}_\Gamma\varphi \mid \mathrm{C}_\Gamma\varphi \mid \overline{\mathrm{C}}_\Gamma\varphi.$$

The temporal operators U and R are named as usual *until* and *release* respectively, X is the next step operator. The operator K_q represents "agent q knows" and $\overline{\mathrm{K}}_q$ is the corresponding dual representing "agent q does not know whether or not something holds". The epistemic operators $\mathrm{D}_\Gamma, \mathrm{E}_\Gamma$, and C_Γ represent distributed knowledge in the group Γ, "everyone in Γ knows", and common knowledge among agents in Γ, respectively. $\overline{\mathrm{D}}_\Gamma, \overline{\mathrm{E}}_\Gamma$, and $\overline{\mathrm{C}}_\Gamma$ are the corresponding dual ones.

Let M be a model, and ρ be a path or run; $\rho(i)$ denote the i-th state of $\rho \in \Pi$, and $\rho[m]$ denote the path or run ρ with a designated formula evaluation position m, where $m \leq length(\rho)$. Note that $\rho[0] = \rho$. Further, let $\Gamma \subseteq \mathcal{A}$. The union of Γ's epistemic indistinguishability relations is defined as $\sim_\Gamma^E = \bigcup_{q \in \Gamma} \sim_q$, $\sim_\Gamma^C$ denotes the transitive closure of $\sim_\Gamma^E$, whereas $\sim_\Gamma^D = \bigcap_{q \in \Gamma} \sim_q$. Then a LTLK formula φ is true along the path ρ (in symbols $M, \rho \models \varphi$) iff $M, \rho[0] \models \varphi$, where

- $M, \rho[m] \models p$ iff $p \in \mathcal{V}(\rho(m))$;
- $M, \rho[m] \models \neg\alpha$ iff $M, \rho[m] \not\models \alpha$;
- $M, \rho[m] \models \alpha \wedge \beta$ iff $M, \rho[m] \models \alpha$ and $M, \rho[m] \models \beta$;
- $M, \rho[m] \models \alpha \vee \beta$ iff $M, \rho[m] \models \alpha$ or $M, \rho[m] \models \beta$;
- $M, \rho[m] \models \mathrm{X}\alpha$ iff $length(\rho) > m$ and $M, \rho[m+1] \models \alpha$;
- $M, \rho[m] \models \alpha\mathrm{U}\beta$ iff $(\exists k \geq m)[M, \rho[k] \models \beta$ and $(\forall m \leq j < k)\ M, \rho[j] \models \alpha]$;
- $M, \rho[m] \models \alpha\mathrm{R}\beta$ iff $[\rho \in \Pi^\omega(\iota)$ and $(\forall k \geq m)\ M, \rho[k] \models \beta]$ or $(\exists k \geq m)[M, \rho[k] \models \alpha$ and $(\forall m \leq j \leq k)\ M, \rho[j] \models \beta]$;
- $M, \rho[m] \models \mathrm{K}_q\alpha$ iff $(\forall \rho' \in \Pi^\omega(\iota))(\forall k \geq 0)[\rho'(k) \sim_q \rho(m)$ implies $M, \rho'[k] \models \alpha]$;
- $M, \rho[m] \models \overline{\mathrm{K}}_q\alpha$ iff $(\exists \rho' \in \Pi(\iota))(\exists k \geq 0)[\rho'(k) \sim_q \rho(m)$ and $M, \rho'[k] \models \alpha]$;
- $M, \rho[m] \models \mathrm{Y}_\Gamma\alpha$ iff $(\forall \rho' \in \Pi^\omega(\iota))(\forall k \geq 0)[\rho'(k) \sim_\Gamma^{\mathrm{Y}} \rho(m)$ implies $M, \rho'[k] \models \alpha]$,
- $M, \rho[m] \models \overline{\mathrm{Y}}_\Gamma\alpha$ iff $(\exists \rho' \in \Pi(\iota))(\exists k \geq 0)[\rho'(k) \sim_\Gamma^{\mathrm{Y}} \rho(m)$ and $M, \rho'[k] \models \alpha]$, where $\mathrm{Y} \in \{\mathrm{D}, \mathrm{E}, \mathrm{C}\}$.

Given a global state g of M and an LTLK formula φ, we use the following notations. $M, g \models \varphi$ iff $M, \rho \models \varphi$ for all the paths $\rho \in \Pi(g)$. $M \models \varphi$ iff $M, \iota \models \varphi$. $M, g \models^\exists \varphi$ iff $M, \rho \models \varphi$ for some path $\rho \in \Pi(g)$. $Props(\varphi)$ is the set of atomic propositions appearing in φ. LTL is the sublogic of LTLK which consists only of the formulae built without epistemic operators. ELTLK is the existential fragment of LTLK, defined by the following grammar: $\varphi ::= p \mid \neg p \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathrm{X}\varphi \mid \varphi \mathrm{U} \varphi \mid \varphi \mathrm{R} \varphi \mid \overline{\mathrm{K}}_i \varphi \mid \overline{\mathrm{E}}_\Gamma \varphi \mid \overline{\mathrm{D}}_\Gamma \varphi \mid \overline{\mathrm{C}}_\Gamma \varphi$. An ELTLK formula φ holds in the model M, denoted $M \models^\exists \varphi$, iff $M, \rho \models \varphi$ for some path or run $\rho \in \Pi(\iota)$. The intuition behind this definition is that ELTLK is obtained only by restricting the syntax of the epistemic operators while the temporal ones remain the same. We get the existential version of these operators by the change from the universal ($\models$) to the existential quantification ($\models^\exists$) over the paths in the definition of the validity in the model M. Notice that this change is only necessary when φ contains a temporal operator, which is not nested in an epistemic operator. Our semantics meets two important properties. Firstly, for LTL the definition of validity in a model M uses paths only. Secondly, if we replace each Π with Π^ω, the semantics does not change as our models have total transition relations (each run is a prefix of some path). The semantics applied to submodels of M does not have the above property, but it preserves ELTLK over M, which is shown in Theorem 2.

3 BMC for ELTLK

SAT-Based BMC. In this section we present a SAT-based BMC method for ELTLK, which is based on ideas from [1,14]. We present only the translation to propositional formulae, since the bounded semantics that is the backbone of the proposed translation can be easily derived from this translation. We start by defining k-paths, (k,l)-*loops*, and the function $f_k : \mathrm{ELTLK} \to \mathbb{N}$ that gives a bound on the number of k-paths, which are sufficient to validate a given ELTLK formula.

Let $M = (G, \iota, \Pi, \{\sim_q\}_{q \in \mathcal{A}}, \mathcal{V})$ be a model, $k \geq 0$ a bound, and φ an ELTLK formula. A k-path is the prefix of length k of a path in Π. A k-path ρ is the (k,l)-*loop* iff $\rho(l) = \rho(k)$, for some $0 \leq l < k$; note that (k,l)-loop ρ generates the inifinite path of the following form: $\rho = u \cdot v^\omega$ with $u = (\rho(0), \ldots, \rho(l))$ and $v = (\rho(l+1), \ldots, \rho(k))$. To calculate the value of $f_k(\varphi)$, we first extend the formula φ to the formula $\varphi' = \mathrm{E}\varphi$. Next, we calculate the value of f_k for φ' in the following way. if $p \in \mathcal{PV}$ then $f_k(p) = f_k(\neg p) = 0$, $f_k(\mathrm{E}\varphi) = f_k(\varphi) + 1$, $f_k(\varphi \vee \psi) = max\{f_k(\varphi), f_k(\psi)\}$, $f_k(\varphi \wedge \psi) = f_k(\varphi) + f_k(\psi)$, $f_k(\mathrm{X}\varphi) = f_k(\varphi)$, $f_k(\varphi \mathrm{U} \psi) = k \cdot f_k(\varphi) + f_k(\psi)$, $f_k(\varphi \mathrm{R} \psi) = (k+1) \cdot f_k(\psi) + f_k(\varphi)$, $f_k(\overline{\mathrm{C}}_\Gamma \varphi) = f_k(\varphi) + k$, $f_k(\overline{Y}\varphi) = f_k(\varphi) + 1$ for $\overline{Y} \in \{\overline{\mathrm{K}}_i, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{E}}_\Gamma\}$.

The problem of checking whether M is a model for φ can be translated to the problem of checking the satisfiability of the following propositional formula: $[M,\varphi]_k := [M^{\varphi,\iota}]_k \wedge [\varphi]_{M,k}$. The formula $[M^{\varphi,\iota}]_k$ constrains the $f_k(\varphi)$ symbolic k-paths to be valid k-paths of M, while the formula $[\varphi]_{M,k}$ encodes a number of constraints that must be satisfied on these sets of k-paths for φ to be satisfied. Once this translation is defined, checking satisfiability of an ELTLK formula can be done by means of a SAT-solver.

In order to define the formula $[M,\varphi]_k$, we proceed as follows. We begin with the encoding of the global states in the model M; recall that the set of global states G is a subset of the product of the agents local states. We assume that $L_i \subseteq \{0,1\}^{nl_i}$ with $nl_i = \lceil log_2(|L_i|) \rceil$, and $\sum_{i=1}^{n} nl_i = m$ for some $m \in N$. So, each global state $g = (l_1, \ldots, l_n) = (g[1], \ldots, g[m])$ of M can be represented by a vector $w = (w[1], \ldots, w[m])$ (called *symbolic state*), where each $w[i] \in \mathcal{PV}$, for $i = 1, \ldots, m$, is a propositional variable (called state variable). A finite sequence $(w_0, \ldots, w_k)$ of symbolic states is called a *symbolic k-path*. In general, we need to consider not just one, as for pure LTL, but a number of symbolic k-paths. This number depends on the formula φ under investigation, and it is given by the value $f_k(\varphi)$. The j-th symbolic k-path is denoted by $w_{0,j}, \ldots, w_{k,j}$, where $w_{i,j}$ is a symbolic state for $0 \leq j < f_k(\varphi)$, and $0 \leq i \leq k$. For two symbolic states w, w', we define the following auxiliary propositional formulae: $I_g(w)$ - encodes the global state g of the model M, $p(w)$ - encodes a set of states where the proposition $p \in \mathcal{PV}$ holds, $H_q(w, w')$ - encodes equivalence of local states of agent q, $\mathcal{R}(w, w')$ - encodes the transition relation of M, $L_{l,k}(j)$ - encodes a (k, l)-loop on the j-th symbolic k-path.

Let $w_{i,j}$ be symbolic states for $0 \leq i \leq k$ and $0 \leq j < f_k(\varphi)$. The propositional formula $[M^{\varphi,\iota}]_k$ is defined as follows: $[M^{\varphi,\iota}]_k := \bigwedge_{j=0}^{f_k(\varphi)-1} \bigwedge_{i=0}^{k-1} I_\iota(w_{0,j}) \wedge \mathcal{R}(w_{i,j}, w_{i+1,j})$. The next step is a translation of an ELTLK formula φ to the propositional formula (1): $[\varphi]_{M,k} := [\varphi]_k^{[0,0]} \vee \bigvee_{l=0}^{k-1} (L_{l,k}(0) \wedge [\varphi]_{k,l}^{[0,0]})$, where $[\varphi]_k^{[m,n]}$ denotes the translation of φ at the symbolic state $w_{m,n}$ using the n-th symbolic k-path, $[\varphi]_{k,l}^{[m,n]}$ denotes the translation of φ at the symbolic state $w_{m,n}$ using the n-th symbolic k-path which is a (k, l)-loop; note that in Formula 1 we either do not put any assumption on the shape of all the $f_k(\varphi)$ symbolic $k-$paths, or we require that all the $f_k(\varphi)$ symbolic $k-$paths represent $(k, l)-$loops. Such a definition is fully consistent with the definition of the bounded semantics.

Definition 1 (Translation). *Let φ be an ELTLK formula, $k \geq 0$ a bound, and $0 \leq m, l \leq k$. Moreover, let $succ(m) = m+1$ for $m < k$, and $succ(m) = l+1$ for $m = k$.*

- $[p]_k^{[m,n]} := p(w_{m,n})$, • $[p]_{k,l}^{[m,n]} := p(w_{m,n})$,
- $[\neg p]_k^{[m,n]} := \neg p(w_{m,n})$, • $[\neg p]_{k,l}^{[m,n]} := \neg p(w_{m,n})$,
- $[\alpha \wedge \beta]_k^{[m,n]} := [\alpha]_k^{[m,n]} \wedge [\beta]_k^{[m,n]}$, • $[\alpha \wedge \beta]_{k,l}^{[m,n]} := [\alpha]_{k,l}^{[m,n]} \wedge [\beta]_{k,l}^{[m,n]}$,
- $[\alpha \vee \beta]_k^{[m,n]} := [\alpha]_k^{[m,n]} \vee [\beta]_k^{[m,n]}$, • $[\alpha \vee \beta]_{k,l}^{[m,n]} := [\alpha]_{k,l}^{[m,n]} \vee [\beta]_{k,l}^{[m,n]}$,
- $[\mathrm{X}\alpha]_k^{[m,n]} :=$ *if* $m < k$ *then* $[\alpha]_k^{[m+1,n]}$ *else* $false$, • $[\mathrm{X}\alpha]_{k,l}^{[m,n]} := [\alpha]_{k,l}^{[succ(m),n]}$,
- $[\alpha \mathrm{U} \beta]_k^{[m,n]} := \bigvee_{i=m}^{k}([\beta]_k^{[i,n]} \wedge \bigwedge_{j=m}^{i-1}[\alpha]_k^{[j,n]})$, • $[\alpha \mathrm{U} \beta]_{k,l}^{[m,n]} := \bigvee_{i=m}^{k}([\beta]_{k,l}^{[i,n]} \wedge \bigwedge_{j=m}^{i-1}[\alpha]_{k,l}^{[j,n]}) \vee \bigvee_{i=l+1}^{m-1}([\beta]_{k,l}^{[i,n]} \wedge \bigwedge_{j=m}^{k}[\alpha]_{k,l}^{[j,n]} \wedge \bigwedge_{j=l}^{i-1}[\alpha]_{k,l}^{[j,n]})$,
- $[\alpha \mathrm{R} \beta]_k^{[m,n]} := \bigvee_{i=m}^{k}([\alpha]_k^{[i,n]} \wedge \bigwedge_{j=m}^{i}[\beta]_k^{[j,n]})$, • $[\alpha \mathrm{R} \beta]_{k,l}^{[m,n]} := \bigwedge_{i=min(l,m)}^{k}[\beta]_{k,l}^{[i,n]} \vee \bigvee_{i=m}^{k}([\alpha]_{k,l}^{[i,n]} \wedge \bigwedge_{j=m}^{i}[\beta]_{k,l}^{[j,n]}) \vee \bigvee_{i=l+1}^{m-1}([\alpha]_{k,l}^{[i,n]} \wedge \bigwedge_{j=m}^{k}[\beta]_{k,l}^{[j,n]} \wedge \bigwedge_{j=l}^{i}[\beta]_{k,l}^{[j,n]})$,
- $[\overline{\mathrm{K}}_q\alpha]_k^{[m,n]} := \bigvee_{t=0}^{f_k(\varphi)-1} \bigvee_{j=0}^{k}([\alpha]_k^{[j,t]} \wedge H_q(w_{m,n}, w_{j,t}))$,
- $[\overline{\mathrm{K}}_q\alpha]_{k,l}^{[m,n]} := \bigvee_{t=0}^{f_k(\varphi)-1} \bigvee_{j=0}^{k}(\bigvee_{ll=0}^{k-1}([\alpha]_{k,ll}^{[j,t]} \wedge L_{ll,k}(t)) \wedge H_q(w_{m,n}, w_{j,t}))$,
- $[\overline{\mathrm{D}}_\Gamma\alpha]_k^{[m,n]} := \bigvee_{t=0}^{f_k(\varphi)-1} \bigvee_{j=0}^{k}([\alpha]_k^{[j,t]} \wedge \bigwedge_{q\in\Gamma} H_q(w_{m,n}, w_{j,t}))$,
- $[\overline{\mathrm{D}}_\Gamma\alpha]_{k,l}^{[m,n]} := \bigvee_{t=0}^{f_k(\varphi)-1} \bigvee_{j=0}^{k}(\bigvee_{ll=0}^{k-1}([\alpha]_{k,ll}^{[j,t]} \wedge L_{ll,k}(t)) \wedge \bigwedge_{q\in\Gamma} H_q(w_{m,n}, w_{j,t}))$,
- $[\overline{\mathrm{E}}_\Gamma\alpha]_k^{[m,n]} := \bigvee_{t=0}^{f_k(\varphi)-1} \bigvee_{j=0}^{k}([\alpha]_k^{[j,t]} \wedge \bigvee_{q\in\Gamma} H_q(w_{m,n}, w_{j,t}))$,
- $[\overline{\mathrm{E}}_\Gamma\alpha]_{k,l}^{[m,n]} := \bigvee_{t=0}^{f_k(\varphi)-1} \bigvee_{j=0}^{k}(\bigvee_{ll=0}^{k-1}([\alpha]_{k,ll}^{[j,t]} \wedge L_{ll,k}(t)) \wedge \bigvee_{q\in\Gamma} H_q(w_{m,n}, w_{j,t}))$,
- $[\overline{\mathrm{C}}_\Gamma\alpha]_k^{[m,n]} := [\bigvee_{i=1}^{k}(\overline{\mathrm{E}}_\Gamma)^i\alpha]_k^{[m,n]}$, • $[\overline{\mathrm{C}}_\Gamma\alpha]_{k,l}^{[m,n]} := [\bigvee_{i=1}^{k}(\overline{\mathrm{E}}_\Gamma)^i\alpha]_{k,l}^{[m,n]}$.

Now consider a definition of the formula $[\varphi]_{M,k}$ in which we require the set of $f_k(\varphi)$ symbolic $k-$paths to represent strictly either a set of $(k,l)-$loops, or a set of $k-$paths which are not $(k,l)-$loops for any $l < k$, i.e., the following definition (2): $[\varphi]_{M,k} := ([\varphi]_k^{[0,0]} \wedge \neg L_k) \vee \bigvee_{l=0}^{k-1}(L_{l,k}(0) \wedge [\varphi]_{k,l}^{[0,0]})$, with $L_k := \bigwedge_{j=0}^{f_k(\varphi)-1}(\bigvee_{l=0}^{k-1} L_{l,k}(j))$; note that Formula 2 strictly corresponds to the second conjunct of the formula defining the general translation of the BMC-method for LTL (see Def. 9 of [1]). We have observed that this additional "no loop" constraint embedded in the translation implemented according to Formula 2 does not help the SAT-solver MiniSat 2 to check the resulting formula more efficiently than in the case of the translation implemented according to Formula 1. The corresponding experimental results are presented in Section 4.

The correctness of the translation scheme is guaranteed by the following theorem.

Theorem 1. *Let M be a model, and φ an ELTLK formula. Then, $M \models^{\exists} \varphi$ iff there exists $k \geq 0$ such that $[\varphi]_{M,k} \wedge [M^{\varphi,\iota}]_k$ is satisfiable.*

Proof. In the full version of the paper[1] omitted due to a lack of space.

BDD-Based BMC. Here we explain how to perform BMC of ELTLK using BDDs [5] by combining the standard approach for ELTL [4] with the method for the epistemic operators [15] in a similar manner to the solution for CTL* of [5].

For an ELTLK formula φ we define inductively the *number* $\gamma(\varphi)$ *of nested epistemic operators* in the formula: $\gamma(\varphi) = 0$ if $\varphi = p$ where $p \in \mathcal{PV}$; $\gamma(\varphi) = \gamma(\varphi')$ if $\varphi = \odot\varphi'$ and $\odot \in \{\neg, \mathrm{X}\}$; $\gamma(\varphi) = \gamma(\varphi')+\gamma(\varphi'')$ if $\varphi = \varphi' \odot \varphi''$ and $\odot \in \{\wedge, \vee, \mathrm{U}, \mathrm{R}\}$; $\gamma(\varphi) = \gamma(\varphi') + 1$ if $\varphi = \mathrm{Y}\varphi'$ and $\mathrm{Y} \in \{\overline{\mathrm{K}}_q, \overline{\mathrm{E}}_\Gamma, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{C}}_\Gamma\}$. Let $\mathrm{Y} \in \{\overline{\mathrm{K}}_q, \overline{\mathrm{E}}_\Gamma, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{C}}_\Gamma\}$. If $\varphi = \mathrm{Y}\psi$ is an ELTLK formula, by $sub(\varphi)$ we denote the formula ψ nested in the epistemic operator Y. For an arbitrary ELTLK formula φ we define inductively the set $\mathcal{Y}(\varphi)$ of its subformulae in the form $\mathrm{Y}\psi$: $\mathcal{Y}(\varphi) = \emptyset$ if $\varphi = p$ where $p \in \mathcal{PV}$; $\mathcal{Y}(\varphi) = \mathcal{Y}(\varphi')$ if $\varphi = \odot\varphi'$ and $\odot \in \{\neg, \mathrm{X}\}$; $\mathcal{Y}(\varphi) = \mathcal{Y}(\varphi') \cup \mathcal{Y}(\varphi'')$ if $\varphi = \varphi' \odot \varphi''$ and $\odot \in \{\wedge, \vee, \mathrm{U}, \mathrm{R}\}$; $\mathcal{Y}(\varphi) = \mathcal{Y}(\varphi') \cup \{\varphi\}$ if $\varphi = \mathrm{Y}\varphi'$ and $\mathrm{Y} \in \{\overline{\mathrm{K}}_q, \overline{\mathrm{E}}_\Gamma, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{C}}_\Gamma\}$. Let $M = (G, \iota, \Pi, \{\sim_q\}_{q\in\mathcal{A}}, \mathcal{V})$ and $U \subseteq G$ with $\iota \in U$. The *submodel* generated by U is a tuple $M|_U = (U, \iota, \Pi', \{\sim'_q\}_{q\in\mathcal{A}}, \mathcal{V}')$, where: $\sim'_q = \sim_q \cap U^2$ for each $q \in \mathcal{A}$, $\mathcal{V}' = \mathcal{V} \cap U^2$, and Π' is the set of the paths and runs of M having all the states in U, formally, $\Pi' = \{\rho \in \Pi \mid (\forall 0 \leq i \leq length(\rho))\ \rho(i) \in U\}$. For ELTLK formulae φ, ψ, and ψ', by $\varphi[\psi \leftarrow \psi']$ we denote the formula φ in which every occurrence of ψ is replaced with ψ'. By $\mathcal{V}_M$ we understand the function $\mathcal{V}$ of the model M, and by $G_R \subseteq G$ the set of its reachable states. Moreover, we define $[\![M, \varphi]\!] = \{g \in G_R \mid M, g \models^{\exists} \varphi\}$. Given a model M and an ELTLK formula φ, Algorithm 1 is used to compute the set $[\![M, \varphi]\!]$, under the assumption that we have the algorithms for computing this set for each φ being an ELTL formula or in the form $\mathrm{Y}p$, where $p \in \mathcal{PV}$, and $\mathrm{Y} \in \{\overline{\mathrm{K}}_q, \overline{\mathrm{E}}_\Gamma, \overline{\mathrm{D}}_\Gamma, \overline{\mathrm{C}}_\Gamma\}$ (we use the algorithms from [4] and [15], respectively). In order to obtain this set, we construct a new model M_c together with an ELTL formula φ_c, as described in Algorithm 1, and compute the set $[\![M_c, \varphi_c]\!]$, which is equal to $[\![M, \varphi]\!]$. Initially φ_c equals φ, which is an ELTLK formula, and we process the formula in stages to reduce it to an ELTL formula by replacing with atomic propositions all its subformulae containing epistemic operators. We begin by choosing some

[1] Available at http://verics.ipipan.waw.pl/r/1

epistemic subformula ψ of φ_c, which consists of exactly one epistemic operator, and process it in two stages. First, we modify the valuation function of M_c such that every state initialising some path or run along which $sub(\psi)$ holds is labelled with the new atomic proposition $p_{sub(\psi)}$, and we replace with the variable $p_{sub(\psi)}$ every occurrence of $sub(\psi)$ in ψ. In the second stage, we deal with the epistemic operators having in their scopes atomic propositions only. By modifying the valuation function of M_c we label every state initialising some path or run along which the modified simple epistemic formula ψ holds with a new variable p_ψ. Similarly to the previous stage, we replace every occurrence of ψ in φ_c with p_ψ. In the subsequent iterations, we process every remaining epistemic subformulae of φ_c in the same way until there are no more nested epistemic operators in φ_c, i.e., we obtain an ELTL formula φ_c, and the model M_c with the appropriately modified valuation function. Finally, we compute the set of all reachable states of M_c that initialise at least one path or run along which φ_c holds (line 13).

Algorithm 1. Computing $[\![M, \varphi]\!]$

1: $M_c := M, \varphi_c := \varphi$
2: **while** $\gamma(\varphi_c) \neq 0$ **do**
3: pick $\psi \in \mathcal{Y}(\varphi_c)$ such that $\gamma(\psi) = 1$
4: **for all** $g \in [\![M_c, sub(\psi)]\!]$ **do**
5: $\mathcal{V}_{M_c}(g) := \mathcal{V}_{M_c}(g) \cup \{p_{sub(\psi)}\}$
6: **end for**
7: $\psi := \psi[sub(\psi) \leftarrow p_{sub(\psi)}]$
8: **for all** $g \in [\![M_c, \psi]\!]$ **do**
9: $\mathcal{V}_{M_c}(g) := \mathcal{V}_{M_c}(g) \cup \{p_\psi\}$
10: **end for**
11: $\varphi_c := \varphi_c[\psi \leftarrow p_\psi]$
12: **end while**
13: **return** $[\![M_c, \varphi_c]\!]$

Algorithm 2. The BMC algorithm for ELTLK

1: $Reach := \{\iota\}, New := \{\iota\}$
2: **while** $New \neq \emptyset$ **do**
3: $Next := New_{\rightsquigarrow}$
4: **if** $\iota \in [\![M|_{Reach}, \varphi]\!]$ **then**
5: **return** $true$
6: **end if**
7: $New := Next \setminus Reach$
8: $Reach := Reach \cup New$
9: **end while**
10: **return** $false$

To perform bounded model checking of an ELTLK formula, we use Algorithm 2. Given a model M and an ELTLK formula φ, the algorithm checks if there exists a path or run initialised in ι on which φ holds, i.e., if $M, \iota \models^{\exists} \varphi$. For any $X \subseteq G$ by $X_{\rightsquigarrow} \stackrel{def}{=} \{g' \in G \mid (\exists g \in X)(\exists \rho \in \Pi(g))\ g' = \rho(1)\}$ we define the set of the immediate successors of all the states in X. The algorithm starts with the set $Reach$ of reachable states that initially contains only the state ι. With each iteration the verified formula is checked (line 4), and the set $Reach$ is extended with new states (line 8). The algorithm operates on submodels $M|_{Reach}$ generated by the set $Reach$ to check if the initial state ι is in the set of states from which there is a path or run on which φ holds. The loop terminates if there is such a path or run in the obtained submodel, and the algorithm returns $true$ (line 4). The search continues until no new states can be reached from the states in $Reach$. When we obtain the set the of reachable states, and a path or run from the initial state on which φ holds could not be found in any of the obtained submodels, the algorithm terminates with $false$. The correctness of the results obtained by the BMC algorithm is formulated in the following theorem:

Theorem 2. *Let $M = (G, \iota, \Pi, \{\sim_q\}_{q \in \mathcal{A}}, \mathcal{V})$ be a model, φ an ELTLK formula, and $\rho \in \Pi$ a path or run with an evaluation position m such that $m \trianglelefteq_\rho length(\rho)$. Then, $M, \rho[m] \models \varphi$ iff exists $G' \subseteq G$ such that $\iota \in G'$, and $M|_{G'}, \rho[m] \models \varphi$.*

Proof. In the full version of the paper:[1] omitted due to a lack of space.

To define the sets of states corresponding to the ELTL formulae that are needed in Algorithm 1, we use the method described in [4] based on the idea of checking the non-emptiness of Büchi automata. As we are unable to include more details (due to the page limit), we refer the reader to [4]. The method we use for ELTL has some limitations when used for BMC, where it is preferable to detect counterexamples using not only the paths but also the runs of the submodel. As the method assumes totality of the transition relation of the verified model, it allows finding counterexamples only along the paths of the model. However, this does not change the correctness of the results if the final submodel has the total transition relation: in the worst case the detection of the counterexample is delayed to the last iteration, i.e., when all the reachable states are computed. Nonetheless this should not keep us from assessing the potential efficiency of the approach, as if the method used for computing the set of states for an ELTL formula would be able to detect counterexamples sooner.

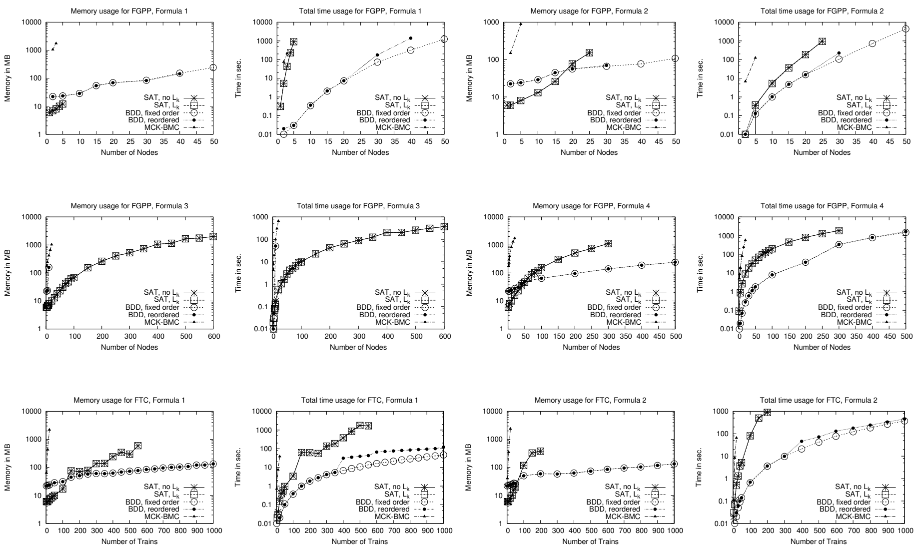
4 Experimental Results and Final Remarks

The two methods have been implemented as prototype modules of the tool VerICS [11]. We consider three scalable systems for which we give performance evaluation of our BMC algorithms and the BMC algorithm of MCK for the verification of properties expressed in LTLK. The following systems are considered: Faulty Generic Pipeline Paradigm (FGPP), Faulty Train Controller (FTC), and Dining Cryptographers (DC). Details on the performed benchmarks together with the verified formulae are available at the webpage of VerICS.[2]

Our methods of BMC for LTLK using SAT and BDDs are, to our best knowledge, the first ones formally presented in the literature. However, at the time of writing this paper, we have received a new version of the tool MCK from its authors. The manual for MCK states that the tool supports SAT-based BMC for CTL*K (i.e., CTL* extended with the epistemic component). Unfortunately, no theory behind this implementation has ever been published. We are aware of the paper [9], which describes SAT-based BMC for CTLK, but it does not discuss how this approach can be extended to CTL*K. Therefore, we cannot compare our SAT-based BMC method for LTLK with the one for CTL*K implemented in MCK. On the other hand, we can compare our experimental results with these of MCK as LTLK is a subset of CTL*K. We ensured that for each considered benchmark, the counterexamples found by the tools are of similar size, i.e., either they are constant or their complexity is the same with respect to the number of the processes. The tests have been performed on a computer with 2 GHz processor, with the limits of 2 GB of memory and 2000 seconds.

An important difference in performance between the SAT- and BDD-based BMC reveals itself in the FTC benchmark, where the BDD-based method performs much better in terms of the total time and memory consumption. In the case of FGPP, BDD-BMC is still more efficient, but the difference is not that significant. Our SAT-based BMC significantly outperforms the BDD-based BMC for φ_2 of DC: SAT-BMC has computed

[2] `http://verics.ipipan.waw.pl/benchmarks`



the results for 3500 cryptographers, whereas BDD-BMC for 41. The reason is that there are at most two symbolic k-paths, and the length of the counterexamples is constant. It is also the case for φ_3 of FGPP. The efficiency of BDD-BMC improves for the formula φ_4 of FGPP comparing to φ_3, although they are similar. The reason is the presence of the knowledge operator that causes the partitioning of the problem to several smaller ELTL verification problems, which are handled much better by the implementation of the operations on BDDs. A noticeable superiority of SAT-BMC for φ_2 of DC follows from the long encoding times of the BDD for the transition relation. The reordering of the BDD variables reduces the memory consumption in the case of DC. This means that the fixed interleaving order we used can often be considered optimal, but the loss in the verification time to reorder the variables, in favour of reducing memory consumption, is also not significant and is often worth the tradeoff. In the case of φ_3 for DC, SAT-BMC was remarkably inferior to BDD-BMC, i.e., SAT-BMC managed to compute the results only for 3 cryptographers in the time of 5400 seconds, whereas BDD-BMC managed to compute the results for 17 cryptographers. This follows from the fact that φ_3 contains the common knowledge operator, which requires many symbolic k-paths to be analysed. We do not include the graph for this case. For φ_1 of DC, our BDD-BMC has computed the results for 14 cryptographers, outperforming SAT-BMC (4 cryptographers).

In most cases, BDD-BMC spends a considerable amount of time on encoding the system, whereas SAT-BMC on verifying the formula. Therefore, BDD-BMC may provide additional time gains when verifying multiple specifications of the same system.

We include comparisons of MCK with our methods for the cases where the lengths of counterexamples scale correspondingly, thus minimising the factor played by different semantics. The comparison shows that for FGPP and FTC our BDD-BMC and SAT-BMC are superior to MCK for all the tested formulae (sometimes by several orders of magnitude). There could be several reasons for this. While our approach is especially optimised for LTLK, it is likely that MCK treats LTLK formulae as CTL*K formulae, for which the translation is typically much less efficient. MCK consumes all the available memory even when the formulae are surprisingly small (approx. 10^6 clauses and 10^5 variables) compared to those successfully tested in our SAT-BMC experiments (more than 10^8 clauses and variables in some cases). However, it should be noted that MCK implements different semantics of MAS, in which agents can perform independent actions simultaneously in a single step of the protocol, what may result in different counterexamples than given by IIS. This is the case of the DC benchmark, where MCK can profit from the strong locality and produces counterexamples of constant length, independently of the number of cryptographers, for all the formulae, being able to verify 15, 32, and 14 cryptographers for φ_1, φ_2, and φ_3, respectively. Using our approaches, we could verify, respectively, 14 cryptographers (BDD-BMC), 3500 (SAT-BMC), 41 (BDD-BMC). From our analysis we can conclude that our two BMC approches remain complementary and none of them is clearly superior in general, whereas in most cases MCK seems to be inferior to our BMC approaches.

References

1. Biere, A., Cimatti, A., Clarke, E., Strichman, O., Zhu, Y.: Bounded model checking. In: Highly Dependable Software. Advances in Computers, vol. 58. Academic Press (2003)
2. Bordini, R.H., Fisher, M., Pardavila, C., Visser, W., Wooldridge, M.: Model Checking Multi-Agent Programs with CASP. In: Hunt Jr., W.A., Somenzi, F. (eds.) CAV 2003. LNCS, vol. 2725, pp. 110–113. Springer, Heidelberg (2003)
3. Cabodi, G., Camurati, P., Quer, S.: Can BDD compete with SAT solvers on bounded model checking? In: DAC 2002, pp. 117–122 (2002)
4. Clarke, E., Grumberg, O., Hamaguchi, K.: Another Look at LTL Model Checking. In: Dill, D.L. (ed.) CAV 1994. LNCS, vol. 818, pp. 415–427. Springer, Heidelberg (1994)
5. Clarke, E., Grumberg, O., Peled, D.: Model Checking. MIT Press (1999)
6. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.: Reasoning about Knowledge. MIT Press, Cambridge (1995)
7. Gammie, P., van der Meyden, R.: MCK: Model Checking the Logic of Knowledge. In: Alur, R., Peled, D.A. (eds.) CAV 2004. LNCS, vol. 3114, pp. 479–483. Springer, Heidelberg (2004)
8. van der Hoek, W., Wooldridge, M.: Model Checking Knowledge and Time. In: Bošnački, D., Leue, S. (eds.) SPIN 2002. LNCS, vol. 2318, pp. 95–111. Springer, Heidelberg (2002)
9. Huang, X., Luo, C., van der Meyden, R.: Improved Bounded Model Checking for a Fair Branching-Time Temporal Epistemic Logic. In: van der Meyden, R., Smaus, J.-G. (eds.) MoChArt 2010. LNCS (LNAI), vol. 6572, pp. 95–111. Springer, Heidelberg (2011)
10. Jones, A., Lomuscio, A.: A BDD-based BMC approach for the verification of multi-agent systems. In: CS&P 2009, vol. 1, pp. 253–264. Warsaw University (2009)
11. Kacprzak, M., Nabiałek, W., Niewiadomski, A., Penczek, W., Półrola, A., Szreter, M., Woźna, B., Zbrzezny, A.: VerICS 2007 - a Model Checker for Knowledge and Real-Time. Fundamenta Informaticae 85(1-4), 313–328 (2008)
12. Lomuscio, A., Penczek, W., Qu, H.: Partial order reduction for model checking interleaved multi-agent systems. In: AAMAS, pp. 659–666. IFAAMAS Press (2010)
13. van der Meyden, R., Shilov, N.V.: Model Checking Knowledge and Time in Systems with Perfect Recall. In: Pandu Rangan, C., Raman, V., Sarukkai, S. (eds.) FST TCS 1999. LNCS, vol. 1738, pp. 432–445. Springer, Heidelberg (1999)
14. Penczek, W., Lomuscio, A.: Verifying epistemic properties of multi-agent systems via bounded model checking. Fundamenta Informaticae 55(2), 167–185 (2003)
15. Raimondi, F., Lomuscio, A.: Automatic verification of multi-agent systems by model checking via OBDDs. Journal of Applied Logic 5(2), 235–251 (2007)

Modelling Trusted Web Applications

Mohammed Y. Alzahrani and Lilia Georgieva

School of Mathematical and Computer Sciences,
Heriot-Watt University, Edinburgh, EH14 4AS, UK
{lilia,mya9}@hw.ac.uk

Abstract. In this paper we model message exchange in a trusted multi-agent system (MAS). Modelling such MAS allows us to study and analyse agents' behaviour. We apply our model in context where secure agent communication is paramount, i.e. within the context of data-sensitive and time-sensitive web-applications. As a result we aim to reduce design errors of such applications, analyse agents' behaviour in secure and compromised web applications and consequently improve security, reliability and usability.

We use model checking to study the pattern of message exchange among agents. As a verification technique, model checking assists in finding flaws and simplifying the design of an application. We simulate a secure transaction between a client and a server. Our model is realistic; we augment it by introducing discrete time, web-session management, and authentication.

Keywords: Web applications, Model Checking, Multi-Agent Systems.

1 Introduction

We define a multi-agent system (MAS) as a system composed of multiple cooperating autonomous agents which either share information or perform a (sequence of) actions towards a goal or both. An *open* MAS (e.g. a system in which agents communicate over the internet) is a MAS in which agents are operating in uncertain environments, which can expose them to risk. Agents in such environments, which are owned by different stakeholders with different intentions, can freely join and leave at any time. As a result, trust and security can be difficult to maintain; an agent may believe other agents who are providing false information and/or other agents may try to compromise and target a system. Investigating an agent behaviour in various scenarios can assist in evaluating the level of trust and security.

In this paper we study and analyse a model of a MAS in the context of a secure web application. The web application is modelled as a two-timed finite state machine representing the web pages and the internal states. Our contribution is the simulation and analysis of a realistic and trusted client-server communication, taking in concern the notion of discrete time, web-session management, and authentication.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 524–533, 2012.

2 Background

Sectors like banking, shopping and government services rely on web applications to promote and deliver their services [15,20]. The increased use of web applications raises concerns over potential security flaws [16]. The data that web applications handle is often sensitive with respect to both users and service providers. Vulnerabilities of web applications, which may lead to compromise of sensitive information are being reported continuously [10,17].

A study carried out by [10] showed that 75% of online banking web sites have a major security flaw. In 2010, more than 8000 online banking clients' credentials were stolen from a server where they were stored as plain text [11]. As a result development of methods of ensuring security and high design quality of web applications are essential. Usability of web applications needs to also be taken into account. Frequently there is a trade-off between usability and security. Improving the security by complicating authentication by using, for example, an external hardware device, or introducing chip and pin, could be challenging for some users, undoubtedly reducing the usability of the application. The secure access to the web application should be effective, the additional details provided by the users should be learnable and memorable. At the same time inadvertent or targeted malicious behaviour should be limited and measures need to be in place to prevent unwanted actions.

Online security tasks are usually perceived as secondary by most users [27]. Users may view security as an interference to carry on their main goals. For example, using the latest authentication device introduced by HSBC, i.e. "Secure Key device" is challenging for some elderly users because of the relatively small size of the device and the complex levels of authentication.

Designing reliable and secure web applications without reducing the usability requires techniques that allow designers to examine the behaviour of a web application in different scenarios. When employed to establish security, traditional verification techniques such as testing and simulation can miss errors, since they only explore a part of the possible behaviours of the application [1,9]. Theorem provers and proof checkers do not have this shortcoming, but mastering them is time consuming and relies on interaction with the user, which requires user intervention and advanced knowledge.

In this paper we use model checking [5] for fully-automatic analysis of multi-agent message exchange in secure web applications. Given a possible behaviour of a web application and its desirable behaviour; a model checker carries out a verification task and if an error is found, the model checker shows under which circumstances the error is generated. The complexity of model checking web applications is due to the interaction among many different components such as browsers, databases, and proxy servers. Furthermore, a web application has a short development life cycle and often is used by untrained users that could use it unpredictably. Therefore, model checking could provide high quality applications through analysis and verification.

In order to realistically investigate the behaviour of the web application, we integrate discrete time in our model. We simulate a common message exchange protocol, namely, the SSL 3.0 handshake protocol which at the start of the session represents authentication. The SSL 3.0 protocol is widely used worldwide, for example by The Cooperative Bank, The Central Bank USA, the online bank Smile, to ensure that data transmission online is secure. SSL 3.0 is expected to provide strong server authentication in client and server systems, however poor usability results in weak authentication in practice [13].

A feature of many online banking web applications that use SSL 3.0 is *time-out*. Timeout occurs when a message between the communicating parties does not arrive within pre-fixed time (typically in online banking this time limit is set to two minutes). In such cases data retransmissions or automatic logout, leading to the need for the user to re-enter her details have to be considered [6].

To capture this feature of realistic online banking web application we integrate discrete time in our model. Using discrete time in our model allows us to represent web session management. When model checking timed models, using discrete time is preferred to using real time as discrete time reduces the risk of *the state space explosion* [25].

The following security properties should be considered in communication between agents in online banking: *Authentication*: defined as the assurance that the communication originates from its claimant. *Confidentiality*: defined as the assurance that the transmitted data in not accessible to unauthorised parties. *Integrity*: defined as the assurance that the transmitted data cannot be manipulated by parties without being noticed. *Availability*: defined as the assurance that communication reaches its targeted recipient in a timely manner. *Non-repudiation:* defined as the assurance that the originating agent can be held responsible for its communication.

Our approach is applicable to a wider context of multi-agent systems where a set of rules may exist to govern the agents' behaviour from an initial to a desired state. Modelling and model-checking agents' behaviour in such multi-agent system allows for detecting consistency and completeness of the rules and for effective support of design goals, e.g. communication, service access, and identity management.

3 The Use of Formal Methods for Modelling of Trusted and Secure MAS

Formal methods which support the design, implementation and verification of trustworthiness multi-agent systems are invaluable when ensuring that these security properties are preserved. Model checking [5] has been successfully used for verification of MAS [26,28]. Although model checking techniques have been used widely for the verification of hardware systems, they have been increasingly used for the verification communication, security protocols and software development [5,14].

Model checking provides a fully automatic analysis and verification of initial systems designs; by constructing a model of a system and its desirable behaviour a model checker tool can carry out a verification task and if an error is found, the tool shows under which circumstances the error can be generated.

In this paper we model the multi-agent dialogue as a security protocol in order to capture and investigate possible behaviour of a trusted multi-agent system.

Semantic verification is a highly-desirable property for an inter-operable agent system in many theoretical models of multi-agent dialogues [26,2,22]. For example, in the BDI model [2], the dialogue emerges from a sequence of communicative acts performed by an agent to satisfy their intentions. There is frequently underlying sincerity assumption in such MAS which asserts that agents always act in accordance with their intentions. This assumption is considered restrictive and unreasonable in an open environment with respect to safety and security guarantees.

In our approach the desirable behaviour of the web application is written in a formal language which captures the requirement specification. We simulate a realistic client/server communication and model an online banking web application. We do not make additional explicit sincerity assumptions. We study and model authentication, web session management, and web navigation behaviour.

In order to investigate the behaviour of our model, we take in account the modelling of *time* and the security protocol used [3]. We integrate discrete time and simulate a security protocol at the start of the session to represent the authentication.

3.1 SPIN and Promela

SPIN and its input language Promela developed by [14] for the verification and analysis of distributed software systems making them suitable for simulating and verifying web applications. The models written in Promela (a Process Meta Language), demonstrate the system components as processes that can communicate between each other via channels, earthier by buffered message exchange or rendezvous operations, and also through shared memory represented as global variables [1,14]. SPIN has a successful applications in simulating and formal verifying security protocols and web applications [15,19]. SPIN does not support the modelling *time*, consequently we extend our model with discrete time macros.

3.2 Modelling of Navigation Properties

First we express navigation and security properties in Promela which are crucial for web applications design. The following are examples of such properties: The *h*ome page is reachable from all pages. An agent (user) can logoff at any stage of the transition. The *a*ccount page is reachable from all pages. A page reachable from the *h*ome page or *a*ccount page always has a next page in the transition. An agent (user) can not reach her *account page* without going through login page by providing correct credentials first. Every page is reachable from the *a*ccount page. After a successful *login* all pages are reachable from the *a*ccount page.

4 Adding Realism to Our Model

We model a realistic MAS using discrete time. Discrete time gives us the possibility to construct models of web applications' in which the sequence of steps that the participating agents take in the dialogue (message exchange) are well-defined (discrete).

Modelling discrete time allows us to represent a web session management and possibility to simulate timeout scenario in a web application. In model checking timed models, discrete time is preferred to reduce the risk of *the state space explosion* [25] which is a model checking technique drawback. In contrast, modelling with real time could result to an increase to the system's states to an intractable level. Current model checking tools [14,24] have a better integrated support for discrete time.

Our model closely resembles a real-life process. In a typical exchange with an online banking web application a client is to identify herself before she can proceed with the transaction. The role of discrete time is crucial to identifying the sequences of actions that succeed the authentication.

In a compromised model, for example, if the attacker is active and attempts to impersonate the client, the sequence of actions modelled by the time stamp will be different than the intended one. Consider, for example, the presence of a man in the middle. During such attack the model will be more complex than the secure one and the sequences of actions will be altered. For example, there may be a delay in the client accessing her bank account which in our case will be modelled by a discrete time value which is reflecting the delay.

Time in the discrete time model is divided into equal length slices, indexed by natural numbers.Actions are then framed into those slices; this way it will be possible to measure the elapsed time between events in different slices.

Time ordering of the actions in Promela is implicit and given by the sequential composition of statements. This is similar to ordinary procedural programming languages. However, the time relation is only qualitative, meaning that we only know that an action will eventually be executed after another, but we don't know the exact time interval that will elapse between the two events. This is considered as a significant shortcoming when the correct functioning of the application closely depends on the timing parameters when its behaviour is simulated or verified.

For example in online banking web application where the communication protocols have to deal with unreliable transport media, where the duration of timeout intervals could be of utmost importance. To overcome this problem of timing formal verification methods are extended with the concept of continuous or discrete time. Usually, this is done by labelling each action with a number representing the time moment it is happening; the continuous (dense) time is modelled by real numbers while the discrete time is represented using integers. Discrete time is effective, satisfactory and easier to treat than real time for a broad class of particle systems and applications [1]. We demonstrate that using discrete time is suitable for online banking web application. We employ an approach developed in [1]. We use the existing capabilities of Promela without

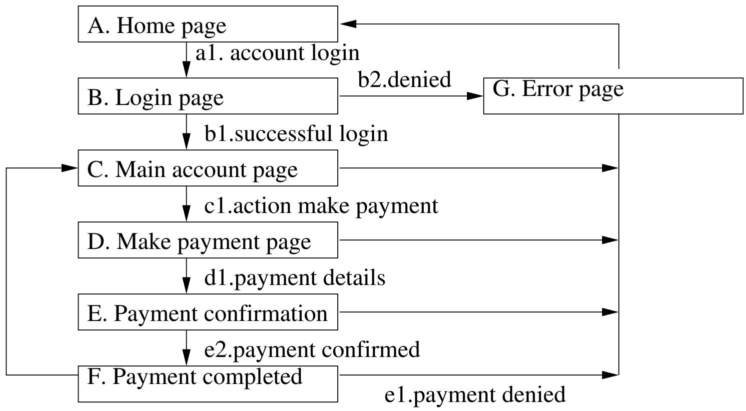


Fig. 1. Web Application Model

any intervention in the software package. Timeout is predefined statement in Promela and can be used as a mechanism for processes synchronization. We introduce new time statements as Promela macro definitions.

In Fig. 1 we show a typical online banking web application. The scenario is of a customer who is logging into her online banking account and carrying out a payment. The transaction starts at the home page, labelled by A and proceeds through B, C, D, E, and F. The transaction is possible only provided that the login is successful. If a login is denied, the customer is redirected to the error page and from there to the home page. Our model is realistic. Consistently with contemporary online banking web sites, we allow for timeout. If the customer has exceeded the allocated time for a transaction, she is redirected to an error page and from there to the home page. Timeout is a safety property, in existing web applications, time out used to ensure safety of the customer's account and is usually preset to two minutes of inactivity.

To build the Promela model of such a web application, the first step is to define the processes representing the roles of each party of model simulation.We define a proctype `client` which describes the behaviour of the initiator of the session with the proctype `server`, (i.e. the customer). The web page transition in our model is represented by a two finite state automata, i.e. of the client and the internal state automata that specifies how the server responds, using the approach developed in [15]. We represent each automaton as a Promela *process* donated by using the keyword `proctype`.

The second step is to simulate a real web session management. We use discrete time as defined in [1]. We successfully model a client/server communication by using multiple clocks that create channel delays and message processing time that result to a timeout state.

For discrete time extension we use the following macro definitions and processes:

```
#define timer int
#define set (tmr,val)  (tmr=val)
#define expire (tmr) (tmr==0) /*timeout*/
#define tick(tmr) if :: tmr>=0 -> tmr=tmr-1 :: else fi
proctype Timers()
{ do :: timeout -> atomic{ tick(tmr1); tick(tmr2) } od }
```

The first macro definition declares a `timer` as a synonym for the integer type. `Set` macro is used to set the value of `tmr` to `val`. The `expire` macro is a test that becomes true if `tmr` becomes `0`. The last macro is `tick` used in the `Timers` process to decrease the value of `tmr` on the condition that it is active (its value is non-negative) . The `Timers` process is the master timing process that control timer by using ticks resulting in decreasing its value. `Timers` use an endless do iteration that realizes the passage of time. It runs concurrently with other process of the system. `Timeout` is predefined Promela statement that becomes true when no other statements within the system is executable. `Timers` process guard `tick`s with timeout to ensure that no process will proceed with an action from the next time slice until the other processes have executed all actions from the current time slice. `Atomic` statement is used to avoid unwanted unblocking of some of the system processes before all timers are decreased.

In any requests to a server, in our model, the client uses a *unique identifier*. In the case of a session-based web application, such as online banking web applications, a session provides a timeout feature that terminates the session on the server side when the maximum period of inactivity has passed. When such a session ends in high sensitive web-based applications (due to timeout), a new session has to be restarted. The restart will include re-login by the user.

In our web banking model we first define a channel for each the client and the server. We then specify time bounds for the acceptable delay, which is in line with the time out feature of online banking websites. The exchange starts with the client being directed to the login page. Authentication takes place next. As in a realistic online banking application, the client is allowed to access his account only provided that he inputs the correct credentials.

In this example scenario, we model the client making a payment to his own bank account. Cases such as withdrawal or transfer of funds are analogous. Provided that the client makes a payment while the session has not expired, he will get a message, confirming that the transaction has taken place. In the case that the client does not identify herself correctly or that the session expires before he does so, the client is directed to an error page.

Fig. 2 shows the sequence chart of our model simulation. Similarly to a realistic online banking application, the first two processes which are instantiated are `proctype Clocks` and `proctype Timers`. Next, `proctype client` sends a request to `proctype server` for login page. The client and server then proceed with the authentication whilst the clock is running. Following successful authentication and provided that no timeout has occurred, the client sends requests to carry out the rest of the transition or exit the application.

The model given in Fig. 2 is a simulation. Simulation of secure web applications is valuable because it allows us to design a simple, yet realistic, models

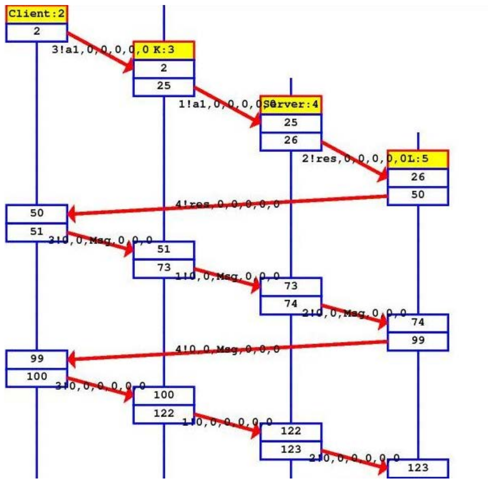


Fig. 2. Simulation Online Banking Web Application Model

and use the existing capabilities of SPIN to extend the model. As our understanding of the processes involved deepens, we are able to extend the model with features such as time, time sequence, and implicitly sequence of actions. Correct sequence of actions is expected if an exchange with a client and an online banking application is not compromised.

Discrete time in our model is used as a time stamp and can give us a reliable indication of the correct succession of actions, for example, no one is allowed to log out before they have logged in or no account information will be disclosed unless correct credentials are provided. Discrete time is a convenient way to model timeout, which is a feature that many realistic bank applications have.

5 Related Work

In this paper we simulate a secure web session control using discrete time and a security protocol using the model checker SPIN. We model web navigation and user interactions using two finite-state automata that represent page transitions and internal web state transitions following an approach developed in [15]. The authentication is added via a communication security protocol simulation at the start of the communication session. Analysing the behaviour of MAS using a specially designed model checking tools is an emerging research.

Similarly to an approach developed in [23], we capture interactions in a multi-agent system that can be used for verification, that is checking that in a given model a predefined task is satisfiable in the presence of time or data constraints. In the context of time-sensitive and data-sensitive web applications such verification during the early phases of development prevents costly rework during the later phases. The approach is applicable to for example, service modelling, business decision making, process modelling.

A related approach for modelling web applications is developed in [12]. It uses communicating finite automata model based on the user-defined properties to be validated. Similarly to our approach, in [8,21] web applications are modelled as pages, links and frames in one state transition. A specifically developed tool ReWeb is used to analyse web sites and understand their organisation in terms of navigation views and variable usage in [8]. Our advantage is that we do not rely on specially developed tools, but use a standard model checker.

An alternative technique for modelling of static and dynamic aspects of web-based applications is given in [29]. The technique is based on locating atomic elements of dynamically created web pages that have static structure and dynamic data contents. These elements are dynamically combined to create composite web pages using sequence, selection, aggregation, and regular expressions. We will explore how the technique can be adapted to our approach in future work.

A framework for modelling of high-level specification of interactive, data-driven web applications is proposed in [7]. In [18] web applications are modelled in terms of the behaviour, object and its structural prospective, and then derived automatically both structural and behaviour test cases.

An approach to assessing the robustness of contracts based on commitments in a multi-agent system is proposed in [4]. We plan to investigate how our approach can be used to model, simulate, and analyse commitment-based contract models in future work.

The work presented in this paper is a simulation of the behaviour of a secure web application, In future work we plan to specify invariants for the model of the web application in LTL and verify properties of this model. In addition, we plan to analyse the differences between a compromised model and a secure model by investigating the sequence of timed actions that occur during a transaction.

References

1. Bosnacki, D., Dams, D.: Integrating real time into SPIN: A prototype implementation. In: FORTE XI / PSTV XVIII, pp. 423–438. Kluwer, B. V. (1998)
2. Bratman, M.: Intention, plans, and practical reason. Harvard University Press (1987)
3. Cheng, Y., Kang, W., Xiao, M.: Model checking of SSL 3.0 protocol based on SPIN. In: 2nd International Conference on Industrial and Information Systems, vol. 2, pp. 401–403. IEEE (2010)
4. Chopra, A.K., Oren, N., Modgil, S., Desai, N., Miles, S., Luck, M., Singh, M.P.: Analyzing Contract Robustness through a Model of Commitments. In: Weyns, D., Gleizes, M.-P. (eds.) AOSE 2010. LNCS, vol. 6788, pp. 17–36. Springer, Heidelberg (2011)
5. Clarke, E.M.: The Birth of Model Checking. In: Grumberg, O., Veith, H. (eds.) 25 Years of Model Checking. LNCS, vol. 5000, pp. 1–26. Springer, Heidelberg (2008)
6. Corin, R., Etalle, S., Hartel, P.H., Mader, A.: Timed analysis of security protocols. Journal of Computer Security 15(6), 619–645 (2007)
7. Deutsch, A., Sui, L., Vianu, V.: Specification and verification of data-driven web applications. Journal of Computer and System Sciences 73(3), 442–474 (2007)

8. Di Sciascio, E., Donini, F.M., Mongiello, M., Piscitelli, G.: Web applications design and maintenance using symbolic model checking. In: Seventh European Conference on Software Maintenance and Reengineering, pp. 63–72. IEEE (2003)
9. Donini, F.M., Mongiello, M., Ruta, M., Totaro, R.: A model checking-based method for verifying web application design. ENTCS 151(2), 19–32 (2006)
10. Falk, L., Prakash, A., Borders, K.: Analyzing websites for user-visible security design flaws. In: Proc. of the 4th Symposium on Usable Privacy and Security, pp. 117–126. ACM, New York (2008)
11. Open Security Fundation. Data loss database annual report (2010), http://datalossdb.org/reports
12. Haydar, M., Petrenko, A., Sahraoui, H.: Formal Verification of Web Applications Modeled by Communicating Automata. In: de Frutos-Escrig, D., Núñez, M. (eds.) FORTE 2004. LNCS, vol. 3235, pp. 115–132. Springer, Heidelberg (2004)
13. Hole, K., Moen, V., Tjostheim, T.: Case study: Online banking security. IEEE Security and Privacy (2006)
14. Holzmann, G.J.: The SPIN model checker: Primer and reference manual. Addison-Wesley Professional (2004)
15. Homma, K., Izumi, S., Takahashi, K., Togashi, A.: Modeling, verification and testing of web applications using model checker. IEICE Transactions on Information and Systems 94(5), 989–999 (2011)
16. Huang, Y.W., Lee, D.T.: Web application security past, present, and future. In: Computer Security in the 21st Century, pp. 183–227 (2005)
17. Jovanovic, N., Kruegel, C., Kirda, E.: Pixy: A static analysis tool for detecting web application vulnerabilities (short paper). In: IEEE Symposium on Security and Privacy (2006)
18. Kung, D.C., Liu, C.H., Hsia, P.: An object-oriented web test model for testing web applications. In: Proc. of the Conference on Quality Software. First Asia-Pacific, pp. 111–120. IEEE (2000)
19. Maggi, P., Sisto, R.: Using SPIN to verify security properties of cryptographic protocols. Model Checking Software, 85–87 (2002)
20. Miao, H., Zeng, H.: Model checking-based verification of web application. Engineering Complex Computer Systems (2007)
21. Ricca, F., Tonella, P.: Analysis and testing of web applications. In: ICSE, pp. 25–34. IEEE Computer Society (2001)
22. Singh, M.: Agent communication languages: Rethinking the principles. IEEE Computer Society (1987)
23. Telang, P., Singh, M.: Specifying and verifying cross-organizational business models: An agent-oriented approach. IEEE Transactions on Services Computing PP(99), 1 (2011)
24. Uppsala University and Aalborg University. Uppaal model checker (2011), http://www.uppaal.org/
25. Valmari, A.: The State Explosion Problem. In: Reisig, W., Rozenberg, G. (eds.) APN 1998. LNCS, vol. 1491, pp. 429–528. Springer, Heidelberg (1998)
26. Walton, C.: Model checking for multi-agent systems. Journal of Applied Logic (Special Issue on Logic-Based Agent Verification) (2006)
27. Whitten, A., Tygar, J.D.: Why johnny can't encrypt: A usability evaluation of pgp 5.0. In: Proceedings of the 8th USENIX Security Symposium, vol. 99 (1999)
28. Wooldridge, M.: Semantic issues in the verification of agent communication languages. Autonomous Agents and Multi-Agent Systems 3(1), 9–31 (2000)
29. Wu, Y., Offutt, J.: Modeling and testing web-based applications. Technical report, George Mason University (2002)

Modelling and Analysis of Dynamic Reconfiguration in BP-Calculus

Faisal Abouzaid[1], John Mullins[1], Manuel Mazzara[2], and Nicola Dragoni[3]

[1] École Polytechnique de Montréal, Campus of the University of Montreal, Canada
{m.abouzaid,john.mullins}@polymtl.ca
[2] Newcastle University, Newcastle upon Tyne, UK and UNU-IIST, Macao
Manuel.Mazzara@ncl.ac.uk
[3] Technical University of Denmark (DTU), Copenhagen
ndra@imm.dtu.dk

Abstract. The BP-calculus is a formalism based on the π-calculus and encoded in WS-BPEL. The BP-calculus is intended to specificaly model and verify Service Oriented Applications. One important feature of SOA is the ability to compose services that may dynamically evolve along runtime. Dynamic reconfiguration of services increases their availability, but puts accordingly, heavy demands for validation, verification, and evaluation. In this paper we formally model and analyze dynamic reconfigurations and their requirements in BP-calculus and show how reconfigurable components can be modeled using handlers that are essential parts of WS-BPEL language.

1 Introduction

In service-oriented computing (SOC), the correct composition of basic services is a key task. The BP-calculus [3] is a π-calculus based formalism encoded in WS-BPEL [15] that has been developed with the intention to provide verification and analysis by mean of model-checking of such compositions.We study here dynamic reconfigurations of BP-processes representing composable services that are self-adaptive and have to react to changes in the environment they are running in. Dynamic reconfiguration provide a mechanism in order to meet these business requirements and a key issue is the preservation of application integrity and correctness; and formal verification must assert that no behavioral invariants get violated through the structural change. The main motivation of this paper is to provide formal foundations to cope with the application integrity and correctness requirement in the context of dynamic reconfiguration.

Although WS-BPEL has not been designed to cope with dynamic reconfiguration, some of its features such as scopes and termination and event handlers can be used for this purpose [12]; and that is the reason why we use the BP-calculus to model dynamic reconfiguration. Requirements that must be insured can then be expressed by mean of the BP-logic (see Section 2.2), opening the way to a formal verification by using existing model-checkers. Once the verification achieved, as BP-processes are encoded in WS-BPEL, it is easy to proceed

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 534–543, 2012.

to the automatic generation of the WS-BPEL code implementing the dynamic reconfiguration. We illustrate our approach on a significant case study drawn from [12].

Related works: The work in [13] motivates the need for a formalism for the modelling and analysis of dynamic reconfiguration of dependable real-time systems. Capabilities of two home-made calculi $Web\pi_\infty$ and CCS_{dp} and those of well-established formalisms namely the asynchronous π-calculus ([8], [4]) and VDM [6] are evaluated.

In [12] and [2] a case study is described using the BPMN notation [5] and formalizations by mean of the $Web\pi_\infty$ and the asynchronous π-calculus are discussed. Finally a BPMN design of the case study is translated to produce a WS-BPEL implementation. However the BPMN notation lacks of formal analysis while the $Web\pi_\infty$ formalism lacks of tools to process automatic verification. Other works exploring how dynamic configuration may be implemented in BPEL are [14] or [11]. The present work can be considered as the continuation of the π-calculus based study, taking advantage of the existence of tools and automatic generation of the corresponding WS-BPEL code.

Another approaches is presented in [9] where the authors use Reo, a channel-based coordination language, to model reconfiguration as a primitive and analyze it using formal verification techniques. A full implementation of the approach in a framework that includes tools for the definition, analysis, and execution of reconfigurations, and is integrated with two execution engines for Reo, is also provided.

Organization of the paper: The rest of the article is organized as follows. In Section 2, we recall the syntax and semantics of the BP-calculus focusing on handlers and the BP-logic. We use this formalism to model dynamic reconfiguration in Section 3. In Section 4 we present the case study and formalize its functional requirements with the BP-logic to illustrate our approach. Finally, Section 5 contains conclusions and directions for future work.

2 The BP-Calculus and the BP-Logic

The main motivation behind the BP-calculus is to provide a rigourous framework for the formal verification and automatic generation of WS-BPEL processes. Its syntax is inspired by the applied π-calculus [1].

We let $\tilde{x} = (x_1, ..., x_n)$, (resp. $\tilde{a} = (a_1, ..., a_m)$, $\tilde{u} = (u_1, ..., u_m)$) range over the infinite set of n-tuples of variables (resp. name, value). We denote $\tilde{x} \leftarrow \tilde{u}$ the assignment of values $\tilde{u}$ to variables $\tilde{x}$. The syntax is summarized in Table 1.

We briefly explain the meaning of the most significant operators together with their intended interpretation. *Serv* denotes the whole defined process (service) and $P, Q, \ldots,$ processes (activities) that may occur within the main service. We syntactically distinguish between them since a whole service may be spawned due to correlation mechanisms while an activity within a process may not. This

Table 1. BP-calculus Syntax

Terms:		
$t ::=$	x	(variables)
	$\mid\ a$	(names)
	$\mid\ u$	(value)
	$\mid\ (t_1, \ldots t_k)$	(tuple)
Correlation Set:		
$\mathcal{C} ::=$	$null \mid \mathcal{C}[\tilde{x} \leftarrow \tilde{u}]$	(correlation set)
Service:		
$Serv ::=$	P	(service)
	$\mid\ [\mathcal{C} : R]c(\tilde{x}).P$	(instance spawn)
$R ::=$	$null \mid R \vert P$	(running instances)
Process:		
$P, Q ::=$	IG	(input guard)
	$\mid\ \overline{c}^t \langle M \rangle.P$	(annotated output)
	$\mid\ \tau.P$	(silent action)
	$\mid\ P \vert Q$	(parallel composition)
	$\mid\ P \triangleright_{c(M)} Q$	(sequential composition)
	$\mid\ if\ M = N\ then\ P\ else\ P$	(conditional)
	$\mid\ S$	(scope)
Guarded choice:		
$IG ::=$	0	(empty process)
	$\mid\ c(u).P$	(input)
	$\mid\ IG + IG'$	(guarded choice)
	$\mid\ [\tilde{x} \leftarrow f(M_1, ..., M_n)]IG$	(function evaluation)
Scopes:		
$S ::=$	$\{\tilde{x},\ P,\ H\}$	(scope)
$H ::=$	$\prod_i W_i(P_{i_1}, \cdots, P_{in_i})$	(handlers)

distinction is conform to the WS-BPEL specification. However, in the sequel, we often use the word "process" for both entities. IG is an input guarded process and $IG + IG'$ behaves like a guarded choice and is intended to be translated by the <pick> element of WS-BPEL language.

$P \triangleright_{c(M)} Q$ expresses a sequential composition from process P passing M to Q (Q can perform actions when P has terminated). M carries binding information between proccesses P and Q. This construct allows to easily mimic the WS-BPEL's element <sequence>.

The other main feature is the definition of scopes. A scope is a wrapper for variables, a primary activity and handlers represented as contexts.

If $S ::= \{\tilde{x},\ P,\ H\}$ is a scope, with handlers $H ::= \prod_{i=0}^{n} W_i(P_{i_1}, \cdots, P_{i_{n_i}})$ then, $\tilde{x}$ are the local variables of the scope, and P its primary activity. H is the parallel composition of handlers W_i and each handler is a wrapper for a tuple of processes $\widehat{P} = (P_1, \ldots, P_n)$ corresponding to the activities the handler has to run when invoked. Finally, $W_i(P_{i1}, \cdots, P_{in_i})$ is the process obtained from the multi-hole context $W_i[\cdot]_1 \cdots [\cdot]_{n_i}$ by replacing each occurrence of $[\cdot]_j$ with P_{ij}.

Handlers are considered as contexts where the designer has only to provide the processes to "fill" the holes and other parameters such as the kind of faults or events that are involved. This approach eases the specification of complex processes.

Interested readers will find a description of operational semantics of the language in Appendix A.

2.1 Handlers Semantics

The WS-BPEL specification defines four sorts of handlers. The **Fault Handler** catches faults signaled by the `<Throw>` element. The **Event Handler** defines the activities relative events such as an incoming message or a timeout. The **Compensation Handler** contains the activity to be performed in the case where under certain error conditions; some previous actions need to be undone. The **Termination Handler** is run after termination of the scope's primary activity and all running event handler instances.

Handlers semantics is widely inspired and adapted from [10].

Note that communication channels between processes ($throw$, en_i, dis_i) are bound names to the whole system.

Fault Handler: Given a tuple of faults ($\tilde{x}$) related to a tuple of processes $\widehat{P} = (P_1, \ldots, P_n)$ the code for the fault handler is:

$$W_{FH}(\widehat{P}) \ ::= \ en_{fh}().\Big(\sum_i \big(x_i(\tilde{y}).(\overline{throw}^{inv}\langle\rangle \mid P_i)\big).\big(\overline{y_1}^{inv}\langle\rangle \mid \overline{y}_{fh}^{inv}\langle\rangle\big) + dis_{fh}()\Big)$$

The en_{fh} channel enables the fault handler. The handler uses a guarded sum to execute an activity P_i, associated with the triggered fault (i). It signals its termination to the activating process on the channel y_1 and to the scope on channel y_{fh}. If necessary, the fault handler is disabled using dis_{fh} channel.

Event Handler: Given a tuple of events ($\tilde{x}$) related to a tuple of processes $\widehat{P} = (P_1, \ldots, P_n)$ the code for the event handler is:

$$W_{EH}(\widehat{P}) \ ::= \ (\nu\tilde{x})\ en_{eh}().\Big(\prod_i \big(!x_i(\tilde{y}).\overline{z}_i\langle\tilde{y}\rangle\big) + dis_{eh}()) \mid \prod_i (z_i(\tilde{u}).P_i)\Big)$$

The event handler is enabled by en_{eh} channel. It then waits for a set of events on channels ($\tilde{x}$) each of them associated with an event. When the event occurs, the associated activity P_i is triggered by a synchronization on channel z_i. The event handler is disabled using dis_{eh} channel.

Compensation Handler: Let P_1 be the scope activity and P_C the compensation activity, the compensation handler is modeled by:

$$W_{CH}(P_1, P_C) \ ::= \ en_{ch}().\Big(z(\tilde{y}).(CC(P_1,\tilde{y}) \mid \overline{throw}^{inv}\langle\rangle) + inst_{ch}().(z(\tilde{y}).P_C \mid \overline{y}_{ch}^{inv}\langle\rangle)\Big)$$

where: $CC(P_1, \tilde{y}) = \prod_{z' \in S_n(P_1)} \overline{z'}^{inv}\langle\tilde{y}\rangle$ compensate children scopes (through channels in $S_n(P_1)$) of activity P_1.

A compensation handler associated with a scope z is first installed at the beginning of the scope through an input on channel $inst_{ch}$ ($\overline{y}_{ch}^{inv}$ signals this installation); it then process its compensation activity P_C. If the compensation handler is invoked but not installed, it signals the termination of the scope activity through channel *throw* and performs children compensation (CC).

Termination Handler: To force the termination of a scope, we first disable it's event handlers and terminate primary activity and all running event handler instances. Then the custom or default `<terminationHandler>` for the scope, is run. The formal model is as follows:

$$W_{TH}(P_T) \ ::= \ term(\tilde{u}).\big(\overline{dis}_{eh}^{inv}\langle\rangle \mid \overline{o}^{inv}\langle\tilde{y}\rangle \mid (P_T \mid \overline{throw}^{inv}\langle\rangle)\big)$$

A termination handler is invoked by the terminating scope using channel *term*. It disables the event handler using channel dis_{eh} and terminates scope's primary activity using channel o. Then custom or default termination process P_T is run.

Scope: Finally, putting all this together leads to the following semantics where the scope is represented by a hole context. Only the scope activity A must be provided by the designer.

$Scope \ ::= (\nu \ throw, en_{eh}, en_{fh}, en_{ch}, dis_{fh}, inst_{ch}, dis_{eh}, term)$

$$\begin{aligned}&\big(W_{EH}(\widehat{A_{eh}}) \mid W_{FH}(\widehat{A_{fh}}) \mid W_{CH}(P_1, P_C) \mid W_{TH}(P_T) \mid (\overline{en_{eh}}\langle\ \rangle.\overline{en_{fh}}\langle\ \rangle.\overline{en_{ch}}\langle\ \rangle)\\ &\mid (A|\overline{t}\langle\ \rangle) \mid c().(\overline{dis_{eh}}\langle\ \rangle.\overline{dis_{fh}}\langle\ \rangle \mid \overline{inst_{ch}}\langle\ \rangle.\overline{term}\langle\ \rangle) \mid y_{eh}().y_{fh}().y_{ch}()\\ &\mid (x_z().(\overline{throw}\langle\ \rangle \mid \overline{dis_{fh}}\langle\ \rangle) + t().\overline{c}\langle\ \rangle)\ \big)\end{aligned}$$

where, $(\overline{en_{eh}}\langle\ \rangle.\overline{en_{fh}}\langle\ \rangle.\overline{en_{ch}}\langle\ \rangle)$ enables handlers; $(A|\overline{t}\langle\ \rangle)$ indicates normal termination by an output on channel t ; $c().(\overline{dis_{eh}}\langle\ \rangle.\overline{dis_{fh}}\langle\ \rangle \mid \overline{inst_{ch}}\langle\ \rangle.\overline{term_{ch}}\langle\ \rangle)$ disables event and fault handlers, installs compensation handler and runs termination handler, in case of normal termination. $(x_z().(\overline{throw}\langle\ \rangle \mid \overline{dis_{fh}}\langle\ \rangle) + t().\overline{c}\langle\ \rangle)$ allows the scope to receive a termination signal on x_z from its parents, or terminate normally by receiving a signal on t. Finally, channels $y_{eh}().y_{fh}().y_{ch}()$ are used to indicate termination of handlers; P_T is the termination process and P_C is the compensation activity.

2.2 The BP-Logic

We close these preliminaries by briefly recalling the *BP-logic*. Its syntax is a slight variant of the π-logic [7]. Its semantics is interpreted on labelled $\checkmark$-transition systems and it is given by the following grammar:

$$\phi ::= true \mid \checkmark \mid \ \sim \phi \mid \phi \ \& \ \phi' \mid EX\{\mu\}\phi \mid EF\phi \mid EF\{\chi\}\phi$$

where μ is a BP-calculus action and χ could be μ, $\sim \mu$, or $\bigvee_{i \in I} \mu_i$ and I is a finite set. $EX\{\mu\}\phi$ is the strong next, while $EF\phi$ means that ϕ must be true sometimes in a possible future. The meaning of $EF\{\chi\}\phi$ is that validity of ϕ must be preceded by the occurrence of a sequence of actions χ. Some useful dual operators are defined as usual: $\phi \vee \phi$, $AX\{\mu\}\phi$, $< \mu > \phi$ (weak next), $[\mu]\phi$ (dual of weak next), $AG\phi$ ($AG\{\chi\}$) (always).

Explicit interpretation of the termination predicate ✓ is that P satifies ✓ (denoted P✓) iff P terminates. Additional rules induced by this predicate in the operational semantics are given in Table 3 of Appendix A.

3 Formalizing Dynamic Reconfiguration

3.1 Dynamic Reconfiguration

Dynamic reconfiguration can easily be coped with using handlers [12]. The regions to be reconfigured have to be represented by scopes so that they could contain adequate termination and event handlers, and be triggered or finished in a clean way.

So, each scope (i.e. region) will be associated with appropriate termination and event handlers. These handlers allow the the transition phase to take place: The new region has to be triggered by the event handler while the old region will be then terminated by the termination handler. Event handlers run in parallel with the scope body; so the old region can be terminated separately while the event handler brings the new region into play.

This is formalized by modeling the system $Workflow$ as a sequence of an *entering* region (R_{enter}), followed by the *transition region* (TR), itself followed by the *finishing* region (R_{finish}). Thus, $Workflow = R_{enter} \triangleright TR \triangleright R_{finish}$.

TR is modeled as a scope with an event handler $W_{EH}(P_{E_1})$. In case the *change configuration* event is triggered, the scope corresponding to the new region is launched, while the first activity corresponding to the old region is terminated by calling its termination handler. Let R_{old} and R_{new} be respectively the old and the new region, both in TR. Thus TR is modeled as follows:

$$\begin{aligned}
TR &= \{\tilde{x}, R_{old}, H_1\} \quad where \\
H_1 &= W_{FH}(P_{F_1})|W_{EH}(P_{E_1})|W_{CH}(P_{C_1})|W_{TH}(P_{T_1}) \\
P_{E_1} &= change().R_{new} \\
R_{new} &= \{\tilde{y}, P_{new}, H_2\}. \\
H_2 &= W_{FH}(P_{F_2})|W_{EH}(P_{E_2})|W_{CH}(P_{C_2})|W_{TH}(P_{T_2}) \\
P_{T_2} &= term().R_{old}
\end{aligned}$$

P_{F_i}, P_{E_i}, P_{C_i} and P_{T_i} are activities respectively associated with fault, event, compensation and termination handlers of the i^{th} scope. The main activity of the transition region TR is the old region R_{old} and is processed first, unless the *change* event is triggered, in which case the new region is processed. P_{E_1} is the activity associated with the event handler in charge of triggering R_{new} by mean of channel *change*. The charge of terminating the old region is devoted to process $W_{TH}(P_{T_1}$. R_{new} is itself a scope with a main activity P_{new} and handlers H_2.

This definition acts as a template for any dynamic reconfiguration scheme. Designers need only to fulfil holes, represented by processes P_X in each handler, to adapt it to their needs.

3.2 Expressing Requirements

Requirements can be expressed as structural and/or behavioral properties that ensure the system's consistency. They are invariants of the system and one must ensure that they are not violated. One must ensure, for instance, that whatever is the used procedure, the result is the same and the logical processing order is respected. This is an example of structural and behavioral invariant since it makes assumptions on the state of the system. We may express this property by the following generic formula where x and y are channels that respectively trigger entering and finishing regions:$AG\{\{x().R_{enter}\}true \wedge EF\{y().R_{finish}\}true\}$.

The $EF\{\chi\}\phi$ operator of the BP-logic allows us to express precedence properties. If R_1 and R_2 are some regions that must be processed in this order, one can formalize it like this: $EF\{\{x()R_1\}true\}\ \{y()R_2\}true$.

We use the ✓ operator to express process termination: $R\checkmark$.

4 The Case Study

In order to illustrate our approach, we use here the same case study presented in [12]. This case study describes dynamic reconfiguration of an office workflow for order processing that is commonly found in large and medium-sized organizations.

4.1 Dynamic Reconfiguration of Office Workflow

A given organisation handles its orders from existing customers using a number of activities arranged according to the following procedure:

An order for a product is received from a customer (`Order Receipt`). An inventory check on the availability of the product and a credit check on the customer are performed that use external services (`Evaluation`). If both the checks are positive, the order is accepted for processing; otherwise the order is rejected. If the order is rejected, a notification of rejection is sent to the customer and the workflow terminates. If the order is to be processed, two activities are performed concurrently: the customer is billed for the total cost of the goods ordered plus shipping costs (`Billing`) and the goods are shipped to the customer (`Shipping`). Finally, the order is archived for future reference (`Archiving`) and a notification of successful completion of the order is sent to the customer (`Confirmation`). In addition, for any given order, `Order Receipt` must precede `Evaluation`, which must precede `Rejection` or `Billing` and `Shipping`.

After some time, managers decide to change the order processing procedure, so that `Billing` is performed before `Shipping` (instead of performing the two activities concurrently). During the transition interval from one procedure to the other, some requirements (invariants) must be met and are presented in Section 4.3.

4.2 The Model in BP-Calculus

The R_{enter} region contains order reception and evaluation($Creditcheck$ and $InventoryCheck$ operations) and is not detailed here. In the same manner, the

R_{finish} region contains $Archiving$ and $Confirmation$ and are not detailed here. We focus on the transtion region TR.

To model TR, we only need to provide the structure of processes P_{new}, R_{old}, P_{T_1} and P_{T_2} to complete the template designed in Section 3.1.

$$\begin{aligned} R_{old} &= BillShip().(\overline{Bill}\, customer, order \mid \overline{Ship}\, customer, order) \\ P_{T_1} &= term_1() \\ P_{new} &= BillShip().\overline{term_1}\,.(\overline{Bill}\, customer, order \rhd \overline{Ship}\, customer, order) \\ P_{T_2} &= term_2() \end{aligned}$$

Regions are invoked using the $BillShip$ channel. In the old region, billing and shipping are processed concurrently, while in the new region this is done sequentially. When the new region is invoked, it begins disabling the old region by invoking its termination Handler using channel $term_1$.

4.3 Formalizing Requirements

Key concerns raised from dynamic configuration and discussed in introduction (Section 1) can be formalized in BP-logic. Concerning the case study some of its requirements have been pointed out in [12]. They could be stated as follows:

1. The result of the billing and shipping process for any given order should not be affected by the change in procedure. A BP-logic formula for this requirement is :

$$AG\{\{Billship()\}true \ \wedge \ EF\{\overline{Bill}\, order\}true \wedge EF\{\overline{Ship}\, order\}true\}$$

2. All orders accepted after the change in procedure must be processed according to the new procedure. The formula is then :

$$AG\{\{change()\}true \ \wedge \ EF\{\overline{term_1}\,\}true\}$$

The formula for termination of the whole process is: $Workflow\checkmark$.

5 Discussion and Future Work

In this work, we have shown a high-level approach for modeling and verifying dynamic reconfigurations of WS-BPEL services. We have expressed their requirements in the BP-logic in order to verify them with reliable existing tools such as Mobility Workbench or the HAL Toolkit.

For the future works, we are currently working on the implementation of a complete tool for analysis and verification of WS-BPEL specifications by means of the BP-calculus and the BP-logic. Another long term objective is to formally prove that our language allows for a sound automatic generation of WS-BPEL code.

References

1. Abadi, M., Fournet, C.: Mobile values, new names, and secure communication. In: 28th Symposium on Principles of Programming Languages, pp. 104–115. ACM Press (2001)
2. Abouzaid, F., Bhattacharyya, A., Dragoni, N., Fitzgerald, J.S., Mazzara, M., Zhou, M.: A case study of workflow reconfiguration: Design, modelling, analysis and implementation. School of computing science technical report series, Newcastle upon Tyne: School of Computing Science, University of Newcastle upon Tyne (2011)
3. Abouzaid, F., Mullins, J.: A calculus for generation, verification and refinement of bpel specifications. Electronic Notes in Theoretical Computer Science 200(3), 43–65 (2008)
4. Boudol, G.: Asynchrony and the π calculus. Technical Report 1702, INRIA SophiaAntipolis(1992)
5. BPMN. Bpmn - business process modeling notation, http://www.bpmn.org/
6. Bjorner, D., Jones, C.B. (eds.): The Vienna Development Method: The Meta-Language. LNCS, vol. 61. Springer, Heidelberg (1978)
7. Ferrari, G., Gnesi, S., Montanari, U., Pistore, M., Ristori, G.: Verifying Mobile Processes in the Hal Environment. In: Hu, A.J., Vardi, M.Y. (eds.) CAV 1998. LNCS, vol. 1427, pp. 511–515. Springer, Heidelberg (1998)
8. Honda, K., Tokoro, M.: An Object Calculus for Asynchronous Communication. In: America, P. (ed.) ECOOP 1991. LNCS, vol. 512, pp. 133–147. Springer, Heidelberg (1991)
9. Krause, C., Maraikar, Z., Lazovik, A., Arbab, F.: Modeling dynamic reconfigurations in reo using high-level replacement systems. Science of Computer Programming 76(1), 23–36 (2011)
10. Lucchi, R., Mazzara, M.: A pi-calculus based semantics for ws-bpel. Journal of Logic and Algebraic Programming (2007)
11. Mazzara, M., Abouzaid, F., Dragoni, N., Bhattacharyya, A.: Design, modelling and analysis of a workflow reconfiguration. In: PNSE 2011, Petri Nets and Software Engineering, Newcastle, UK (2011) (invited talk)
12. Mazzara, M., Abouzaid, F., Dragoni, N., Bhattacharyya, A.: Toward Design, Modelling and Analysis of Dynamic Workflow Reconfigurations - A Process Algebra Perspective. In: Carbone, M., Petit, J.-M. (eds.) WS-FM 2011. LNCS, vol. 7176, pp. 64–78. Springer, Heidelberg (2012)
13. Mazzara, M., Bhattacharyya, A.: On modelling and analysis of dynamic reconfiguration of dependable real-time systems. In: Third International Conference on Dependability (DEPEND 2010), Venice/Mestre, Italy. IEEE Computer Society (2010)
14. Mazzara, M., Dragoni, N., Zhou, M.: Implementing workflow reconfiguration in ws-bpel. Journal of Internet Services and Information Security, JISIS (to appear, 2012)
15. Oasis. Web service business process execution language version 2.0 specification, oasis standard (April 2007), http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf

A Operational Semantics of the BP-Calculus

The operational semantics of the BP-calculus is a labelled transition system generated by inference rules given in Table 2.

Table 2. Operational semantics of the BP-calculus

OPEN	$\frac{P \xrightarrow{\overline{a}\langle u\rangle} P' \quad a\neq u}{\{u,P,\emptyset\} \xrightarrow{\overline{a}\langle u\rangle} P'}$	CLOSE	$\frac{P \xrightarrow{a(u)} P' \quad Q \xrightarrow{\overline{a}\langle u\rangle} Q' \quad u\notin fn(P)}{P\mid Q \xrightarrow{\tau} \{u,P'\mid Q',\emptyset\}}$
RES	$\frac{P \xrightarrow{\alpha} P' \quad n\notin fn(\alpha)\cup bn(\alpha)}{\{n,P,\emptyset\}\xrightarrow{\alpha}\{n,P',\emptyset\}}$	TAU	$\frac{}{\tau.P\xrightarrow{\tau}P}$
OUT	$\frac{}{\overline{c}^t\langle M\rangle.P \xrightarrow{\overline{c}\langle M\rangle} P}$	IN	$\frac{}{c(x).P \xrightarrow{c(M)} P\{M/x\}}$
PAR	$\frac{P\xrightarrow{\alpha}P' \quad bn(\alpha)\cap fn(Q)=\emptyset}{P\mid Q\xrightarrow{\alpha}P'\mid Q}$	SYNC	$\frac{P\xrightarrow{\alpha}P' \quad Q\xrightarrow{\overline{\alpha}}Q'}{P\mid Q\xrightarrow{\tau}P'\mid Q'}$
CHOICE	$\frac{IG_i \xrightarrow{\overline{c_i}^t\langle\rangle} P_i \quad i\in\{1,2\}}{IG_1+IG_2\xrightarrow{\alpha}P_i}$	DEF	$\frac{P\{\tilde{y}/\tilde{x}\}\xrightarrow{\alpha}P' \quad A(\tilde{x})=P}{A(\tilde{x})\xrightarrow{\alpha}P'}$
SCO	$\frac{P\xrightarrow{\alpha}P'}{\{x,P,H\}\xrightarrow{\alpha}\{x,P',H\}}$	HAN	$\frac{H\xrightarrow{\alpha}H'}{\{x,P,H\}\xrightarrow{\alpha}\{x,P,H'\}}$
SPAR	$\frac{P\xrightarrow{\alpha}P' \quad Q\xrightarrow{\overline{\alpha}}Q'}{\{x_1,P,H_1\}\mid\{x_2,Q,H_2\}\xrightarrow{\tau}\{x_1,P',H_1\}\mid\{x_2,Q',H_2\}}$		
IFT-M	$\frac{P\xrightarrow{\alpha}P' \quad M=N}{if(M=N)\ then\ P\ else\ Q\xrightarrow{\alpha}P'}$	IFF-M	$\frac{Q\xrightarrow{\alpha}Q' \quad M\neq N}{if\ (M=N)\ then\ P\ else\ Q\xrightarrow{\alpha}Q'}$
EVAL	$\frac{\tilde{M}=f(M_1,\ldots,M_n)\ P\{\tilde{M}/\tilde{x}\}\xrightarrow{\alpha}P'}{[\tilde{x}\leftarrow f(M_1,\ldots,M_n)]P\xrightarrow{\alpha}P'}$		
SEQ1	$\frac{P\xrightarrow{\alpha}P'}{P\rhd_{c(M)}Q\xrightarrow{\alpha}P'\rhd_{c(M)}Q}$	SEQ2	$\frac{P\xrightarrow{\overline{c}\langle M\rangle}0 \quad Q\xrightarrow{c(M)}Q'}{P\rhd_{c(M)}Q\xrightarrow{\tau}Q'}$
C-SP1	$\frac{Serv_0\xrightarrow{\alpha}Serv_0' \quad Serv=c_A(\tilde{x}).A(\tilde{y})}{[\mathcal{C}:R\mid Serv_0]c_A(\tilde{x}).A(\tilde{y})\xrightarrow{\alpha}[\mathcal{C}:R\mid Serv_0']c_A(\tilde{x}).A(\tilde{y})}$		
C-SPT	$\frac{createInstance(M)=true \quad [\tilde{z}\leftarrow\tilde{u}]=correlationPart(M)}{[null{:}0]c_A(\tilde{x}).A(\tilde{y})\xrightarrow{c_A(M)}[[\tilde{z}\leftarrow\tilde{u}]:A(\tilde{u})]c_A(\tilde{x}).A(\tilde{y})}$		
C-SPF	$\frac{createInstance(M)=true \quad [\tilde{z}\leftarrow\tilde{u}]=corrPart(M)\ [\tilde{z}\leftarrow\tilde{u}]\notin\mathcal{C} \quad Serv=c_A(\tilde{x}).A(\tilde{y})}{[\mathcal{C}:R]c_A(\tilde{x}).A(\tilde{y})\xrightarrow{c_A(M)}[\mathcal{C},[\tilde{z}\leftarrow\tilde{u}]:R\mid A(\tilde{u})]c_A(\tilde{x}).A(\tilde{y})}$		

Table 3. Operational semantics induced by ✓ predicate

TER1	$\frac{}{0\checkmark}$	TER2	$\frac{P\checkmark}{\tau.P\checkmark}$	TER3	$\frac{P\checkmark}{c(\tilde{u}).P\checkmark}$	TER4	$\frac{P\checkmark}{\overline{c}^t\langle M\rangle.P\checkmark}$
TER5	$\frac{P\checkmark \quad Q\checkmark}{(P\rhd_{c(M)}Q)\checkmark}$	TER6	$\frac{P\checkmark \quad Q\checkmark}{(P\mid Q)\checkmark}$	TER7	$\frac{P\checkmark}{\{n,P,\emptyset\}\checkmark}$	TER8	$\frac{P\checkmark}{\{\tilde{x},\ P,\ H\}\checkmark}$

Deciding Roles for Efficient Team Formation by Parameter Learning

Dai Hamada and Toshiharu Sugawara

Department of Computer Science and Eng., Waseda University, Tokyo 1698555, Japan
d.hamada@isl.cs.waseda.ac.jp, sugawara@waseda.jp

Abstract. We propose a learning method for efficient team formation by self-interested agents in task oriented domains. Service requests on computer networks have recently been rapidly increasing. To improve the performance of such systems, issues with effective team formation to do tasks has attracted our interest. The main feature of the proposed method is learning from two-sided viewpoints, i.e., team leaders who have the initiative to form teams or team members who work in one of the teams that are solicited. For this purpose, we introduce three parameters to agents so that they can select their roles of being a leader or a member, then an agent can anticipate which other agents should be selected as team members and which team it should join. Our experiments demonstrated that the numbers of tasks executed by successfully generated teams increased by approximately 17% compared with a conventional method.

1 Introduction

Services on the Internet, such as cloud and grid computing, have recently been increasing. These services usually consist of multiple service elements that should be processed efficiently in different nodes in a distributed manner. Even if only one service element is processed with significant delay or fails to be processed, the service cannot be provided. These services can be modeled as coordinated and collaborative activities in multi-agent systems [4]. A service in this model can be expressed as a task consisting of a number of subtasks each of which corresponds to a service element. Then, these subtasks should be allocated to the right agents that have sufficient resources and required functions. Thus, the efficient formation of a team of appropriate agents for required tasks is a key issue in providing timely services.

However, there are a number of difficulties in team formation in these kinds of applications. First, agents have their own resources and functions that are designed by different organizations/developers. Second, which tasks are requested more vary over time because this is a reflection of human activities in the real world. Thus, it is impossible to anticipate what teams of agents are likely to be appropriate in the design phase of the system. Furthermore, the machines where agents are running may asynchronously be replaced by high-performance machines or may stop due to failures. A huge number of tasks may also be requested. Thus, it is urgent that efficient and appropriate team formation in dynamic environments.

This paper proposes a method of learning how efficiently and effectively self-interested, rational agents form teams to complete given tasks. We used a model of

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 544–553, 2012.

team formation in this learning method similar to that proposed in [3], in which an agent autonomously selects the role of a *leader* or a *member* that it should play in a team. One feature of our learning method is the two-sided viewpoints of a leader and a member; the agent learns how many utilities should be allocated to members as dividends when it has the initiative to form a team as a leader and it also learns how many utilities are expected if it joins one of the teams solicited by other possible leaders. Since all agents are self-interested, i.e., interested in increasing their own utilities, this learning can accomplish appropriate dividends for the received utilities as the result of successful team formations, and can thus increase the total number of utilities. This learning enables agents to determine their own roles according to the kinds of tasks in a queue and, if they are members, to select one of teams solicited by other agents for the greatest efficiency.

Another feature is that a leader can learn about appropriate members of a team based on past acceptance ratios, i.e., which agents the leader should solicit as team members for a given task set and how many it should solicit. If a leader solicits more members, a team can be formed with a higher probability of success but as the team is likely to become larger than required, some agents will be wastefully used. Conversely, if solicitations to membership are declined by many agents, the team cannot execute the task set, which means that the corresponding service cannot be provided. Although this decision is important for effective and economical team formation, it has hardly been discussed in previous studies.

We introduced a number of learning parameters to all agents to implement the above-mentioned functionality of learning, expressing expected utilities as a leader and a member, and the expected ratios of joining teams so that they could receive more utilities. This paper is organized as follows. We discuss some related work in Section 2 and the models for team formation in Section 3. Then, we present the proposed method of team formation in which agents change their utility divisions according to the results from team formation and the received dividends. Section 5 explains the experimental evaluation of our method through which we found it could improve system efficiency by more than 17%, compared with a conventional method.

2 Related Work

Since team formation is a key issue in the literature on multi-agent systems, there have been a number of studies for coalition formations and team formations [7,8,10]. For example,[7] proposed an algorithm for coalition formation to allocate task optimally. However, as its computational complexity was exponential, it was difficult to apply to real-world applications. Later in [8], a polynomial-time method of calculating Pareto-optimal solutions is introduced, but the solutions did not result in maximum utilities for the entire system. As it is almost impossible to anticipate the resources needed for tasks or the resources/functions that each agent has in an open system, these algorithms cannot be applied. Our method is polynomial and can adapt to changes in requested tasks and unknown resources/functions in agents.

[1,5] proposed effective (polynomial time) methods for team formation to allocate resources/tasks using reinforcement learning in a tree-structured agent network. As all

the agents in the network learn, from past experiences, the allocation policies for tasks sent to other agents at lower levels of the hierarchy, teams can be formed effectively in their approaches. Although these studies have not assumed the agents' resources and the kinds of requested tasks are known, they have assumed that the agent network is hierarchical; thus, their applications are restricted. [6] obtained the team members necessary to process tasks by negotiations using heuristics. However, its model was simpler than ours, because it expressed whether each task was able to be executed or not in a certain agent.

[3] is related to our proposed learning, in which the method of learning for efficient team formation in task-oriented domain is proposed. In learning, all agents store, for any task T, which agents proposed T and which agents accepted or rejected doing T. Then, the agent acting as a leader selects team members using history information. However, agents in their method only learn which others are likely to accept the team members for the task set containing T but they do not learn which roles of a leader or a member should be played and which roles can increase the received utilities. Thus, their improvements are limited. Furthermore, as the leader agent in their model can select any tasks, after a sufficient learning time, it may only select specific tasks and thus some types of tasks may seldom be selected, which is not realistic in actual applications. Our method can learn what actions are appropriate to increase utilities from the viewpoints of leaders and members. It also chooses the tasks in the order they are requested.

3 Models and Problem Description

3.1 Agents and Tasks

Let $A = \{1, 2, \ldots, n\}$ be a set of agents. Agent i has p kinds of resources that are described by vector $\boldsymbol{h}_i = (h_i^1, \ldots, h_i^p)$, where h_i^k represents the amount of k-th resources and is a non-negative integer. Let $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$ be a set of all kinds of tasks. Task T_i requires various resources to execute it and these resources are denoted by vector $\boldsymbol{R}_{T_i} = (r_{T_i}^1, r_{T_i}^2, \ldots, r_{T_i}^p)$, where $r_{T_i}^j$ is a non-negative integer (and at least, $\exists k$ s.t. $r_{T_i}^k > 0$), expressing the amount of required j-th resources. We also represent T_i's utility, u_{T_i} which agents will receive if T_i is successfully executed. For simplicity, we have assumed that u_{T_i} is the sum of all resources required by T_i, thus $u_{T_i} = \sum_{k=1}^{p} r_{T_i}^k$.

We set $B = \{t_1, \ldots\}$ to be the finite set of tasks, and assume that the same tasks may appear more than once in B. Let $G (\subseteq A)$ be the team of agents that will execute B. We assume that G can execute B if and only if the sum of the k-th resources of all agents in G is equal or larger than the sum of the k-th resources required by all tasks in B:

$$\sum_{i \in G} h_i^k \geq \sum_{t \in B} r_t^k, \tag{1}$$

for $\forall k$ $(1 \leq k \leq p)$, and we can say that the team for B is successfully formed. No agent can belong to multiple teams at the same time.

When the team for B is successfully formed (so it executes the tasks in B), the leader agent receives the sum of the utilities for all tasks, $u_B = \sum_{t \in B} u_t$. The leader of the team collects a certain ratio of utility u_B in return for acting as the leader and then divides the rest of the utilities equally among the other members. The ratio of the returns is learned by each agent so that they can efficiently form a team.

We call the unit of time a *tick*. A number of tasks are added in every tick to the queue; task $t \in \mathcal{T}$ is generated and is added to the tail of the queue with the probability, ρ_t, which is called the task generation probability of t. An agent selects a number of tasks to process from the head of the queue.

3.2 Team Formation

Let $Q =< t_1, t_2, \cdots >$ be the queue of tasks requested by the environment. First, an agent determines its role, as a *leader* or a *member* of the team. An agent that decides to act as a leader voluntarily attempts to form a team for a set of tasks B that are selected from Q. An agent that decides to act as a member selects one of the proposed teams to join. Because all agents are self-interested, they select actions, including role decisions, to (be expected to) maximize utilities. How they determine the role and how many tasks in Q are peeked at will be discussed later.

Leader $l \in A$ selects a number of tasks B from Q and the members of the team it will work with. The set of these members with l is called a *pre-team* in this paper. Next, l sends *member solicitation messages* to the pre-team members and waits for their responses. Suppose that G is the set of l and the agents that answered acceptance messages by l. If G satisfies Formula (1), l informs the members in G that the *team* has successfully been formed. Then, all the agents in G execute the tasks in B and they return to an idle state. If the team cannot satisfy Formula (1), the team formation for B fails and l sends *team abort messages* to the members in G. B is then discarded and the agents in G enter an idle state.

However, an agent that decides to act as a member looks at the member solicitation messages that are received and selects one of these. How this is selected will be discussed in the next section. It then sends its acceptance of joining the team and also sends *member declination messages* to the other leaders. If it does not receive any member solicitation messages or receives team abort messages from the leader, it returns to an idle state.

3.3 Task Set for Execution

Agent i in an idle state first searches tasks from the top of the queue and finds a number of unmarked tasks and marks “peeked” on them. The set of marked tasks is denoted by $B_i = \{t_1, t_2, ..., t_{M_i}\}$. We define the value of M_i as the maximal integer that meets the following condition:

$$\sum_{k=1}^{M_i} \sum_{j=1}^{p} r_{t_k}^j \leq \sum_{j=1}^{p} h_i^j \times L, \tag{2}$$

where L is a positive number, meaning the ratio of the sum of resources of tasks to that of the task, which i can process alone. When there are not a sufficient number of tasks, i will take all the tasks in Q. If $Q = \emptyset$, i tries to act as a member. Then i determines its role by comparing the expected utilities when it acts as a leader with those when it selects one of the member solicitation messages and acts as a member. If it decides to act as a member, the marks for tasks in B_i are deleted.

4 Proposed Learning for Team Formation

Because we assumes that all agents would want to receive more utilities, they would also try to reduce failures in team formation. It is also better to form teams with fewer redundant agent resources to execute B. We introduced three parameters for this purpose, i.e., the *degree of greediness* (GD), the *expected ratio of team solicitation acceptance* (ETSA), and the *expected member utility division ratio* (EMU). Agents learn the values of these parameters through experience to receive more utilities. This section explains the meanings of these parameters and how agents learn their values.

4.1 Degree of Greediness

The degree of greediness (GD) g_i ($0 \leq g_i \leq 1$) of leader agent i is the ratio of the amount of utilities that i collects in return for acting as a leader if i can succeed in forming team G for task set B: Therefore, i collects utility $u_i^{leader} = u_B \times g_i$. The rest of the utilities are allocated equally among all other members:

$$u_j^{member} = \frac{u_B - u^{leader}}{|G| - 1},$$

where $j \in G \setminus \{i\}$.

If g_i is large, the members in the teams formed by leader i cannot expect higher utilities. These members are likely to decline member solicitation messages from i, thus making it difficult for i to act as a leader. As the GD value in each agent depends on the resources it has and on the task-request distribution from the environment, it must be adaptively learned. For this purpose, i updates the value of g_i, depending on the success or failure of a team being formed, using:

$$g_i = \alpha_g \times \delta_{success} + (1 - \alpha_g) \times g_i,$$

where $\delta_{success} = 1$ if a team is successfully formed and $\delta_{success} = 0$, otherwise. Parameter α_g is the learning rate of GD ($0 \leq \alpha_g \leq 1$).

4.2 Expected Ratio of Team Solicitation Acceptance

For agents, i and j, the parameter called the *expected ratio of team solicitation acceptance* (ETSA) of i to j, $e_{i,j}$, expresses the degree of acceptance when i sends a member solicitation message to j. This parameter is used by leader i to select members of the pre-team according to the values of $e_{i,*}$.

More precisely, leader i determines the members of the pre-team for task set B as follows. We initially set $G = \{i\}$. Let $x_B^m = \max(\sum_{t \in B} r_t^m - \sum_{j \in G} e_{i,j} \cdot h_j^m, 0)$. Then, if the following condition holds, the pre-team is the current G:

$$x_B^m = 0 \;\text{ for }\; 1 \leq \forall m \leq p. \tag{3}$$

If condition (3) does not hold, i calculates the weighted expected resources for agent $j \in K_i \setminus G$, $v_j = \sum_{m=1}^{p} (e_{i,j} \cdot h_j^m \cdot x_B^m)$, where $K_i \subset A$ is the set of agents i knows.

Then, one agent $\tilde{j}$ is selected from $K_i \setminus G$ according to the values of $\{v_j\}_{j \in K_i \setminus G}$ using some exploration-or-exploitation strategies (such as, greedy, ε-greedy, or roulette-selection strategies). Note that in our experiments explained below, the roulette selection strategy was used to select members. Therefore, i selects $\tilde{j}(\in K_i \setminus G)$ proportionally to $v_k/(\sum_{j \in K_i \setminus G} v_j)$. Then, $G = G \cup \{\tilde{j}\}$ and this process is repeated until condition (3) holds.

Agent i updates the ETSA of $j(\neq i)$ depending on whether or not j accepts member solicitation messages using:

$$e_{i,j} = \alpha_e \times \delta^j_{accept} + (1 - \alpha_e) \times e_{i,j},$$

where $\delta^j_{accept} = 1$ if j accepts the member solicitation and otherwise $\delta^j_{accept} = 0$.

4.3 Expected Member Utility Division Ratio

For agents i and j $(i \neq j)$, the *expected member utility division ratio* (EMU) of i for j, $d_{i,j}$, expresses the ratio of expected utilities to the utility the team received when i accepted the member solicitation message sent by j. The value of this parameter is used to determine whether an agent should act as a leader or a member and, if it acts as a member, which member solicitation message it should accept using some exploration-or-exploitation strategies. Note that the roulette selection strategy was used in our experiments; Let $S = \{s_1, \ldots, s_l\}$ be the set of received member solicitation messages and $l(s_k)$ be the leader that sent message s_k to i. Then, agent i selects one of the received messages s_k with the probability of

$$\frac{\sum_{t \in B(s_k)} u_t \times d_{i,l(s_k)}}{\sum_{s \in S} \sum_{t \in B(s)} u_t \times d_{i,l(s)}},$$

where $B(s)$ for member solicitation message s is the set of tasks that the proposed team will execute.

After agent i has acted as a member of the team whose leader is j, i updates the value of $d_{i,j}$ according to the utilities, U, received from j:

$$d_{i,j} = \alpha_d \times \frac{U}{\sum_{t \in B(s)} u_t} + (1 - \alpha_d) \times d_{i,j},$$

where s is the member solicitation message that is selected. The α_d is the rate of learning for EMU ($0 \leq \alpha_d \leq 1$). Note that if a team formation fails, $U = 0$, so this value also partly reflects the success ratio of team formation.

4.4 Role Selection

Agent i in an idle state peeks at unmarked tasks in turn from the head of queue Q and selects task set B according to the method described in Section 3.3. Then i calculates the expected utilities of U_i^{leader} and U_i^{member}, when it acts as a leader of B and as a member of the proposed team. These values are calculated as:

$$U_i^{leader} = \sum_{t \in B} u_t \times g_i \quad \text{and} \quad U_i^{member} = \sum_{t \in B(s)} u_t \times d_{i,l(s)},$$

where s is the member solicitation message that is temporally selected. Then, i acts as a leader if $U_i^{leader} \geq U_i^{member}$. Otherwise it acts as a member.

Table 1. Initial Values for Learning Parameters

Parameters	Initial values	Parameters	Initial values
GD ($g_i \forall i \in A$)	0.5	α_g	0.05
ETSA ($e_{i,j} \forall i, j \in A$)	0.5	α_e	0.1
EMU ($d_{i,j} \forall i, j \in A$)	0.5	α_d	0.1

5 Experimental Evaluation

5.1 Experimental Settings

We conducted an experiment in which we investigated how the total number of executed tasks was improved by the proposed learning. In all experiments, we set $L = 2$, $K_i = A$, and the learning parameters summarized in Table 1. The data below are the average values for thirty independent trials based on different random seeds.

To evaluate how effectively teams were formed with our proposed learning method, we compared our results with those obtained by the method in [3], which formed teams based on task similarity: For $\forall t \in B$ (how to select task set B was identical in our experiments), agent i stores sent and received member solicitation messages. Then i, for another agent $j (\neq i)$, counts the number of messages whose sender (leader) is j and task set contains t, J_j^m; the number of the messages accepted by j, J_j^a; and the number of the messages declined by j, J_j^d. Then, i selects the set of agents G as a team member under a certain strategy based on the values of $J_j^m + J_j^a - J_j^d$, where G is the minimum set so that condition (3) is satisfied. The strategy is, e.g., (1) the selection of the first $|G|$ agents or (2) roulette selection based on these values. In the experiment described below, strategy (1) was used (but the experimental results are almost identical with both strategies). More details are explained in [3].

We also compared the results with those derived with a random method in which members were randomly selected until condition (3) held. Note that as the random method did not learn ETSA, $e_{i,j} = 0.5$, which was the initial value. This meant that the resources for members in G were more than double the amount of resources in B. The selection of B was identical to those in the other methods.

5.2 Experimental Results

We generated 50 agents and 50 kinds of tasks (i.e., $|A| = |\mathcal{T}| = 50$) and defined the amount of resources in agents and the amount of resources required by tasks as positive integers randomly selected between 1 and 10. We assumed that there would be two types of resources ($p = 2$). Each task $t \in \mathcal{T}$ was generated with probability $\rho_t = 0.4$ and stored into Q every tick. We recorded the numbers of executed tasks every 25 ticks, from the 1st to 10000th ticks.

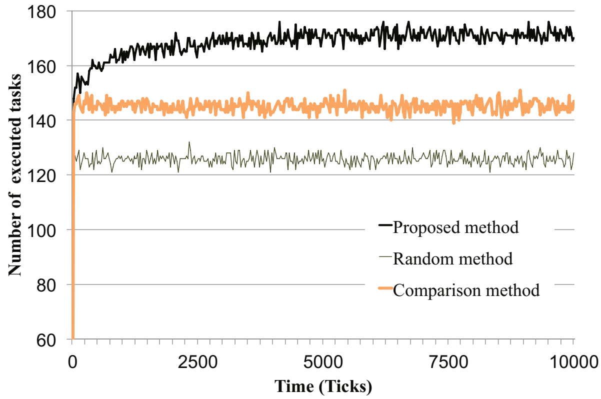


Fig. 1. Transition in numbers of executed tasks in Exp. 1

Figure 1 shows the transitions in the numbers of executed tasks (as the results for successful team formations) over time. This figure shows the numbers of executed tasks with the proposed method are approximately 36.0% higher than those with the random method and 17.2% higher than those with the comparison method.

Although the comparison method also outperformed the random method, its improvements were limited. This was, in part, due to the assumption make for the comparison method; it assumed problem situations in which leader agents could choose any tasks (from the queue) they wanted as elements of B. Therefore,it is possible that some kinds of tasks were not selected after learning. For example, as it was large, it became difficult to form teams. However, agents in our environment had to choose tasks in turn according to the order in which they were requested. Therefore, agents in the comparison method also had to select tasks in turn even though they may not have wanted to select them. However, we believe that our model is more realistic in actual applications.

Table 2. Average sizes and earned utilities

	Proposed method	Comparison method	Random method
Size of successfully formed teams	2.01	1.89	1.70
Earned Utility per 25 ticks	2833.0	2527.5	2064.6

We also investigated the average sizes of successful teams (the number of agents in the team, $|G|$) and the average utility per every 25 ticks earned by all agents. These results are listed in Table 2.

This table reveals that the teams with our method were slightly larger than those with the comparison method and the random method. We considered a task to be small (large) when it required small (large) amount of resources. First, leader agents with the random method seldom succeed team formation and they can only form teams with relatively higher probability when the amount of resources required by B is small and thus the leaders can successfully form teams with small numbers of members. One extreme case is where the team is a single one; this is possible if there is a small task at the head of the task queue but the second task is large. In this case, the task set is small and may require only one or a few agents. Obviously, these situations are inefficient. However, task sets are relatively larger with the comparison method than those with the random method.

However, leaders with the proposed method learn from their local viewpoints, which agents are likely to accept becoming members by estimating the expected acceptance ratio. Then, they select potential members according to the required resources and expected resources provided by other members. Member agents simultaneously learn which leader will distribute greater dividends. Then, according to these learning results from two-sided viewpoints, all agents can effectively determine which role of a leader or a member, can be expected to give them more utilities. These learning based decisions considerably improve the ratio of successful team formations.

5.3 Concluding Remarks

The experiment indicated that our proposed method exhibited high efficiency, resulting in larger numbers of executed tasks, compared with the random and comparison methods. This is because agents that have the enough amounts of required resources have learned to act as leaders. In all the methods used in our experiment, the size of the task set that the agent took for a team was proportional to the amount of its resources. Because larger teams tended to have more utilities, the system performed well if agents with more resources acted as leaders. However, if the task set was large, teams were not likely to be formed because this required more members. The size of the team and the gained utility were relatively larger in ours and the comparison methods than those in the random method. In addition, our method could make these agents act as leaders with a high probability of success.

More precisely, the EMU parameter suggests to each agent which leader's member solicitation message it should select based on the expected utilities, and the GD parameter controls the amount of utilities in return for acting as a leader before the received utilities are divided. However, in the first stage of learning, leader agent i tended to fail in team formations, thus g_i may become smaller. Therefore, many agents tended to act as members. Suppose that a certain agent i that had more resources occasionally acted as a leader and succeeded in forming a team. Parameter g_i is not large and many utilities are divided among all the members. Consequently they may find it easier to accept the member solicitation messages sent by i. Of course, this team formation militates leader agents that have large resources that are requested more. In addition, the ETSA parameter can anticipate which agent is likely to provide their resources and which agent has more scarce resources. This enables the correct selection of member agents. Thus, this combination of learning can improve the overall efficiency of the entire system.

6 Conclusion

We proposed a method of learning how efficiently and effectively self-interested agents form teams for given tasks. All agents learn three parameters for team formation with this method from the viewpoint of both leaders and members. This learning enables agents to decide which role of a leader or a member the agents should play and which agents should be selected as members of a team. We experimentally evaluated the proposed method by comparing it with the conventional and random selection methods. Our learning method could increase the numbers of executed tasks by approximately 17%.

We assumed that agents in our model knew the resources of other agents (although we did not assume that in the design phase of the system). However, some studies such as [2] have not assumed that other agents' resources are known. Instead, agents try to estimate the amount of resources. Therefore, we would like to add this functionality to our method for more general situations. Another direction of this research is, like [9], to extend to the large-scale multi-agent systems where more than hundreds agents work together and the set of K_i is revised based on past experience; this is also another future work. This work was, in part, supported by KAKENHI (22300056).

References

1. Abdallah, S., Lesser, V.: Organization-based cooperative coalition formation. In: Proc. of Int. Conf. on Intelligent Agent Technology, pp. 162–168 (2004)
2. Chalkiadakis, G., Boutilier, C.: Sequential decision making in repeated coalition formation under uncertainty. In: Proceedings of AAMAS 2008, pp. 347–354 (2009)
3. Genin, T., Aknine, S.: Coalition formation strategies for self-interested agents in task oriented domains. In: 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, vol. 2, pp. 205–212 (2010)
4. Huhns, M.N., Singh, M.P., Burstein, M., Decker, K., Durfee, E., Finin, T., Gasser, L., Goradia, H., Jennings, N., Lakkaraju, K., Nakashima, H., Parunak, V., Rosenschein, J.S., Ruvinsky, A., Sukthankar, G., Swarup, S., Sycara, K., Tambe, M., Wagner, T., Zavala, L.: Research directions for service-oriented multiagent systems. IEEE Internet Computing 9, 65–70 (2005)
5. Katayanagi, R., Sugawara, T.: Efficient team formation based on learning and reorganization and influence of communication delay. In: Proceedings of the 11th IEEE International Conference on Computer and Information Technology, CIT 2011 (2011)
6. Kraus, S., Shehory, O., Taase, G.: Coalition formation with uncertain heterogeneous information. In: Proc. of AAMAS 2003 (2003)
7. Shehory, O., Kraus, S.: Methods for task allocation via agent coalition formation. Artificial Intelligence 101(1-2), 165–200 (1998)
8. Shehory, O., Kraus, S.: Feasible formation of coalitions among autonomous agents in nonsuperadditive environments. Computational Intelligence 15(3), 218–251 (1999)
9. Sugawara, T., Hirotsu, T., Kurihara, S., Fukuda, K.: Performance Variation Due to Interference Among a Large Number of Self-Interested Agents. In: Proceedings of 2007 IEEE Congress on Evolutionary Computation, pp. 766–773 (2007)
10. Yang, J., Luo, Z.: Coalition formation mechanism in multi-agent systems based on genetic algorithms. Applied Soft. Computing 7(2), 561–568 (2007)

On the Tight Formation for Multi-agent Dynamical Systems*

Ionela Prodan[1,2], Sorin Olaru[1], Cristina Stoica[1], and Silviu-Iulian Niculescu[2]

[1] SUPELEC Systems Sciences (E3S) - Automatic Control Department, Gif sur Yvette, France
{ionela.prodan,sorin.olaru,cristina.stoica}@supelec.fr
[2] SUPELEC CNRS - Laboratory of Signal and Systems, Gif sur Yvette, France
{ionela.prodan,silviu.niculescu}@lss.supelec.fr

Abstract. This paper addresses the real-time control of multiple agents in the presence of disturbances and non-convex collision avoidance constraints. The goal is to guarantee the convergence towards a tight formation. A single optimal control problem is solved based on a prediction of the future evolution of the system and the resulting controller is implemented in a centralized way. At the supervision level, it is shown that the decision about which agents should take on what role in the desired tight formation is equivalent with a classical pairing (or task assignment) problem. Furthermore, the pairing is re-evaluated at each iteration. The proposed method exhibits effective performance validated through some illustrative examples.

Keywords: Multi-Agent Systems, invariant sets, non-convex constraints, constrained MPC.

1 Introduction

The formation of multiple agents is important in many applications involving the control of cooperative systems [3], as for example, in adversary games applications where the sensor assets are limited. Agent formations allow each member of the team to concentrate its sensors across a specified region of the workspace, while their neighboring members cover the rest of the workspace. Furthermore, in applications like search and rescue, coverage tasks and security patrols, the agents direct their visual and radar search responsibilities depending on their positions in the formation [1]. Other applications include coordinated ocean platform control for a mobile offshore base [2]. The homogeneous modules forming the offshore base must be able to perform long-term station keeping at sea, in the presence of waves, winds and currents. Therefore, the independent modules have to be controlled in order to be maintained aligned (i.e., they converge to a tight formation).

* The research of Ionela Prodan is financially supported by the EADS Corporate Foundation (091-AO09-1006).

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 554–565, 2012.

Furthermore, the problem of maintaining a formation becomes even more challenging if one needs to ensure that all the agents avoid collisions inside the group [14].

There is a large literature dedicated to the formation control for a collection of vehicles using potential field approach [16], or approaches based on graph theory [7]. The authors of [5] and [16] investigate the motions of vehicles modeled as double integrators. Their objective is for the vehicles to achieve a common velocity while avoiding collisions with obstacles and/or agents assumed to be points. The derived control laws involve graph Laplacians for an associated undirected graph and also nonlinear terms resulting from artificial potential functions.

The main goal of this paper is to control a set of agents having independent dynamics while achieving a global objective, such as a tight formation with desired specifications and collision free behavior. For reducing the computation time we use the "nominal" behavior of the agents and consider safety regions around them to compensate for the effects of the disturbances affecting the "real" systems. Further, these regions are defined within the theory of invariant sets in order to avoid recomputations during the real-time functioning. The formation control problem is decomposed in two separate problems:

• The "off-line" definition of the ideal configuration. A minimal configuration is determined with respect to a given cost function under the constraints imposed by the safety regions.

• In real-time, a receding horizon optimization combined with task assignment relative to the minimal configuration will be employed.

The real-time control is designed based on the following two-stage procedure:

1. Determine "who-goes-where" in the formation. This is equivalent with solving a standard assignment problem, which is a special case of the so-called Hitchcock Transportation Problem (TP) [11].
2. Solve a mixed-integer optimization problem according to the target geometry of the formation and the associated safety regions.

Finally, this two separate problems are embedded within a Model Predictive Control (MPC) problem (see, for instance, [9] for basic notions in MPC), leading to an optimization problem for driving the group of agents to a specified formation with associated target locations.

In the present paper, we revise the preliminary results obtained in [12] and introduce enhancements in the control design method which enables the stabilization of the multi-agent formation. We show that for the convergence to the predefined formation an additional fixed point constraint (i.e., the target positions are also equilibrium points for the considered dynamics) must be taken into account. Moreover other contributions of the paper are, on the one hand, the reduction in the computational cost and, on the other hand, the efficient handling of an increased number of constraints.

The rest of the paper is organized as follows. Section 2 describes the individual agents model and the set invariance concept. Furthermore, Section 3 presents the configuration of the desired multi-agent formation. Section 4 states the mixed-integer optimal control problem embedded within MPC. Finally, several concluding remarks are drawn in Section 6 and the illustrative examples are presented in Section 5.

The following notations will be used throughout the paper. Minkowski's addition of two sets $\mathcal{X}$ and $\mathcal{Y}$ is defined as $\mathcal{X} \oplus \mathcal{Y} = \{A + B : A \in \mathcal{X},\ B \in \mathcal{Y}\}$. Let $x_{k+1|k}$ denote the value of x at time instant $k+1$, predicted upon the information available at time $k \in \mathbb{N}$.

2 Preliminaries and Prerequisites

2.1 System Description

Consider a set of N_a linear systems (vehicles, pedestrians or agents in a general form) which model the behavior of individual agents. The i^{th} system is described by the following discrete LTI dynamics affected by additive disturbances:

$$x^i_{k+1_d} = A_i x^i_{k_d} + B_i u^i_{k_d} + w^i_k, \quad i = 1, \dots, N_a, \tag{1}$$

where $x^i_{k_d} \in \mathbb{R}^n$ are the state variables, $u^i_{k_d} \in \mathbb{R}^m$ is the control input and $w^i_k \in \mathbb{R}^n$ represents a bounded disturbance for the agent i. Henceforth we assume the following:

1. The pair (A_i, B_i) is stabilizable, with $A_i \in \mathbb{R}^{n\times n}$, $B_i \in \mathbb{R}^{n\times m}$.
2. The disturbance w^i is bounded, i.e. $w^i \in \mathcal{W}^i$, where $\mathcal{W}^i$ is a convex and compact set containing the origin.

Theoretically, formulation (1) suffices for solving any typical multi-agent control problem (e.g., formation stability, trajectory tracking and so forth). However, the presence of additive noises makes the numerical computation difficult and severely limits the practical implementability. This is particularly true for centralized schemes where the computations are to be made into an extended space.

The solution followed here is based on the ideas in [10]. As a first step, we consider the nominal systems associated to (1):

$$x^i_{k+1} = A_i x^i_k + B_i u^i_k, \quad i = 1, \dots, N_a. \tag{2}$$

By linking the control laws associated to dynamics (1) and (2), respectively, through the relation

$$u^i_{k_d} = u^i_k + K_i(x^i_{k_d} - x^i_k), \tag{3}$$

we observe that the tracking error of the i^{th} system, defined as $z^i_k \triangleq x^i_{k_d} - x^i_k$, is given by:

$$z^i_{k+1} = (A_i + B_i K_i) z^i_k + w^i_k. \tag{4}$$

Assuming that K_i makes the closed-loop state matrix $A_i + B_i K_i$ to be Schur[1], it follows that an RPI (see Definition 1) set S_i can be determined and the "real"

[1] The stabilizability hypothesis on the pair (A_i, B_i) implies the existence of an optimal control law for each agent i, $K_i \in \mathbb{R}^{n\times m}$ such that the matrices $A_i + B_i K_i$ are stable, where the controller K_i, $i = 1, \dots, N_a$ is constructed either by a Linear Quadratic (LQ) design using the solution of the discrete algebraic Riccati equation or alternatively by pole placement technique.

trajectory generated by (1) will reside in a tube centered along the "nominal" trajectory generated by (2):

$$x^i_{k_d} - x^i_k \in S_i \longleftrightarrow x^i_{k_d} \in \{x^i_k\} \oplus S_i \tag{5}$$

as long as[2] $z^i_0 \in S_i$ for any $k \geq 0$.

This permits to consider the nominal system in the subsequent optimization problems and thus minimize the necessary numerical computations.

Definition 1. A set $\Omega \in \mathbb{R}^n$ is *robustly positive invariant* (RPI) for the discrete-time system $x_{k+1} = \Theta x_k + \delta$ with the perturbation $\delta \in \Delta \subset \mathbb{R}^n$, iff $\Theta\Omega \oplus \Delta \subseteq \Omega$.

A set Ω_∞ is *minimal robustly invariant* (mRPI) for some given dynamics iff it is a RPI set in $\mathbb{R}^n$ contained in every RPI set for the given dynamics.

There are various algorithms able to offer arbitrarily close RPI approximation for a mRPI set (as for example, the approaches proposed by [13]). It is worth mentioning that these algorithms ignore the exponentially increase in the complexity of representation.

2.2 Collision Avoidance Formulation

A typical multi-agent problem is the minimization of some cost problem with constraints. As stated before, the original formulation, with dynamics (1) is not optimal since it requires to take into account the bounded disturbances affecting the dynamics. Hereafter we will use the "nominal" dynamics (2) and we will analyze how conditions on the "real" dynamics (1) are transposed to the "nominal" dynamics case.

A classical issue in multi-agent formations is the *collision avoidance*. Using the notation of (1), the condition that any two agents do not collide translates into:

$$\{x^i_d\} \cap \{x^j_d\} = \emptyset, \ \ \forall i, j = 1, \ldots, N_a, \ i \neq j. \tag{6}$$

Using the notation from (2) and assuming that the conditions validating relation (5) are verified, we reach the equivalent formulation:

$$\{x^i \oplus \mathcal{S}_i\} \cap \{x^j \oplus \mathcal{S}_j\} = \oslash, \ \ \forall i, j = 1, \ldots, N_a, i \neq j. \tag{7}$$

This condition takes explicitly (through the use of the sets S_i, S_j) into account the uncertainties introduced by the bounded perturbation in (1).

Let us recall that for any two convex sets $\mathcal{A}, \mathcal{B}$ the following equivalence is true:

$$\mathcal{A} \cap \mathcal{B} = \emptyset \longleftrightarrow 0 \notin \mathcal{A} \oplus \{-\mathcal{B}\}. \tag{8}$$

Using (8), we obtain the equivalent formulation for (7):

$$x^i - x^j \notin S_j \oplus (-S_i), \ \ \forall i, j = 1, \ldots, N_a, \ i \neq j. \tag{9}$$

---

[2] The assumption that the tracking error starts inside the set is made for simplification reasons. As long as the set is contractive, after a finite number of steps, any trajectory starting "outside" will enter inside the set.

Remark 1. The obstacle avoidance problem can be treated in a similar way: $x^i \notin \mathcal{O}_l$, where $\{\mathcal{O}_l\}_{l=1,\dots,M}$ denotes the collection of fixed obstacles (if a particular obstacle is non-convex, it may be seen as a union of convex sets). Then the conditions that have to be verified is:

$$x^i \notin \mathcal{O}_l \oplus (-S_i)\,, \;\; \forall i = 1, \dots, N_a, \; l = 1, \dots, M. \tag{10}$$

Remark 2. Note that, a solution using parametrized polyhedra (see, for instance, [8]) to describe the safety regions of the agents is presented in [12]. For guaranteeing that two (or more) agents do not superpose, the parametrized intersections of the invariant sets are considered and then, the domain for which the intersections are void is described. However, we note that this approach is computationally demanding.

Remark 3. The main technical difficulty encountered in this paper is the fact that, often, the feasible regions are non-convex. This problem rises naturally from separation conditions (see condition (9) and Remark 1). The solution is to use the mixed-integer programming techniques [6]. This allows us to express the original non-convex feasible region as a convex set in an extended space. Such an approach leads to a significant number of binary variables in the problem formulation, thus leading to unrealistic computational times (in the worst-case scenarios, an exponential increase dependent of the number of binary variables). A method for reducing the computational time is detailed in [15], where we propose a technique for making the time of computation P-hard in the number of Linear/Quadratic Programming (LP/QP) subproblems that have to be solved.

3 The Configuration of Multi-agent Formations

3.1 Minimal Configuration of the Multi-agent Formation – Off-Line

The goal of clustering the agents as close as possible to the origin is realized through a minimal configuration for the group of agents (2). We pose the problem as an optimization problem where the cost function is the sum of the square distances of each agent from the origin and the constraints are the ones imposing collision avoidance (9):

$$\min_{(x^i,u^i),\; i=1,\dots,N_a} \sum_{i=1}^{N_a} \|x^i\|_2^2, \quad \text{subject to: } \begin{cases} x^i - x^j \notin S_j \oplus \{-S_i\}\,, \\ x^i = A_i x^i + B_i u^i, \quad \forall i,j = 1,\dots,N_a,\; i \neq j. \end{cases} \tag{11}$$

Solving the mixed-integer optimization problem (11), a set of target positions and the associated control laws are obtained:

$$T = \left\{ (x_f^1, u_f^1), (x_f^2, u_f^2), \dots, (x_f^{N_a}, u_f^{N_a}) \right\}, \tag{12}$$

where every pair (x_f^i, u_f^i) is a fixed state/input of the i^{th} agent.

Remark 4. Note that the second constraint in (11) is a fixed point condition. That is, the optimization problem will find only pairs (x_f^i, u_f^i), $i = 1, \ldots, N_a$ which are also a fixed point for the considered dynamics, (2). Geometrically, this means that the points x_f^i will find themselves on the associated subspaces spanned by $(I_n - A_i)^{-1}B_i$. In particular, if the agents have the same dynamics (i.e., homogeneous agents), they will have a common subspace over which to select the fixed points x_f^i.

3.2 Task Assignment Formulation – On-Line

In the particular case of homogeneous[3] agents (understood here as agents with the same safety regions) we can intercalate an additional step in the control mechanism. Since the agents have the same safety regions, we can change who goes where in the minimal configuration computed in the previous subsection. This is equivalent with finding the best permutation over the set of the final positions in the target formation, T from (12). This is in fact the optimal assignment problem encountered in the field of combinatorial optimization [4].

If one associates a cost with the assignment of agent j to target x_f^i as c_{ij}, the problem of finding the best assignment is defined as:

$$\min_{\delta_{ij},\ i,j=1,\ldots,N_a} \sum_{i=1}^{N_a}\sum_{j=1}^{N_a} c_{ij}\delta_{ij}, \quad \text{subject to: } \begin{cases} \sum\limits_{i=1}^{N_a} \delta_{ij} = 1, \\ \sum\limits_{j=1}^{N_a} \delta_{ij} = 1, \\ \delta_{ij} \in \{0, 1\}, \end{cases} \tag{13}$$

where δ_{ij} are the decision variables: "1" if target x_f^i is assigned to agent j and "0" otherwise. These binary variables ensure that each agent is assigned to one unique target position.

The problem is defined by the choice of the cost weights c_{ij}, the simplest way is to choose it as the distance between the actual position of agent j and the desired target position in the formation. Hence, the problem would be to determine the minimal distance that an agent has to travel to establish the optimal assignment in the specified formation. A more insightful way is to use the unconstrained dynamics (2) of the agents to describe the cost of reaching from the initial position to the desired position. Then, c_{ij} can be described by a weighted norm:

$$c_{ij} = (x^j - x_f^i)^T P (x^j - x_f^i), \quad i, j = 1, \ldots, N_a \tag{14}$$

with the matrix $P = P^T \geq 0$ given by the Lyapunov function or the infinite time cost-to-go, as long as the agents follow the unconstrained optimum through the control action:

$$u^j = -K_j(x^j - x_f^i) + \bar{u}^j, \ \ \forall i, j = 1, \ldots, N_a, \ i \neq j, \tag{15}$$

[3] In the heterogeneous case the reassignment of the final destination points is no longer feasible since the swapping of the safety regions will result in collisions of the agents.

where $\bar{u}^j$ is chosen such that $x_f^j = A_j x_f^i + B_j \bar{u}^j$, with $\bar{u}^j = B_j^{-1}(I - A_j)x_f^i$, if the matrix B_j is invertible (or the alternative pseudo-inverse which allows the definition of a fixed point for the nominal trajectory). This optimization problem can be reduced to a simple LP problem, hence it can be efficiently computed.

4 Receding Horizon Optimization Problem

The goal is to drive the agents to a minimal configuration (12) (if possible, applying optimization (13)) while in the same time avoiding collisions along their evolution towards the formation.

To this end, we will consider the set of N_a constrained systems as a *global system* defined as:

$$\tilde{x}_{k+1} = A_g \tilde{x}_k + B_g \tilde{u}_k, \tag{16}$$

with the corresponding vectors which collects the states and the inputs of each individual nominal system (2) at time k, i.e., $\tilde{x}_k = [x_k^{1\,T} | \cdots | x_k^{N_a\,T}]^T$, $\tilde{u} = [u_k^{1\,T} | \cdots | u_k^{N_a\,T}]^T$ and the matrices which describe the model: $A_g = diag[A_1, \ldots, A_{N_a}]$, $B_g = diag[B_1, \ldots, B_{N_a}]$.

We consider an optimal control problem for the global system where the cost function and the constraints couple the dynamic behavior of the individual agents. Also, perfect knowledge of each agent dynamics described by equation (2) is available to all the other agents. Consequently, the global model will be used in a predictive control context which permits the use of non-convex constraints for collision avoidance behavior.

A finite receding horizon implementation is typically based on the solution of an open-loop optimization problem. An optimal control action $\tilde{u}^*$ is obtained from the control sequence $\tilde{\mathbf{u}} \triangleq \left\{\tilde{u}_{k|k}, \tilde{u}_{k+1|k}, \ldots, \tilde{u}_{k+N-1|k}\right\}$ as a result of the optimization problem:

$$\begin{aligned} \tilde{u}^* &= \arg\min_{\tilde{\mathbf{u}}} V_N(\tilde{x}_{k+1|k}, \ldots, \tilde{x}_{k+N|k}, \tilde{u}_{k|k}, \ldots, \tilde{u}_{k+N-1|k}), \\ \text{subject to: } & \begin{cases} \tilde{x}_{k+l|k} = A_g \tilde{x}_{k+l-1|k} + B_g \tilde{u}_{k+l-1|k}, \ l = 1, \ldots, N \\ H_{ij} \cdot \tilde{x}_{k+l|k} \notin S_j \oplus \{-S_i\}, \ \forall i, j = 1, \ldots, N_a, \quad i \neq j, \end{cases} \end{aligned} \tag{17}$$

where $H_{ij} \triangleq [0 \ \ldots \underbrace{I}_{i} \ \ldots \underbrace{-I}_{j} \ \ldots \ 0]$ is a projection matrix which permits to rewrite the collision avoidance between two agents i and j in the notation of the centralized system (16)[4].

In order to assure that the target positions (12) we require a cost function which is minimized in the destination points (and not the origin):

[4] Using the elements provided in Remark 3, the computational complexity of (17) can be assessed to a polynomial number of QP problems.

$$V_N(\tilde{x}_{k|k}, \tilde{u}_{k|k}) = (\tilde{x}_{k+N|k} - x_f^{i(k)})^T \tilde{P}(\xi_{k+N|k} - x_f^{i(k)}) + \sum_{l=1}^{N-1} (\tilde{x}_{k+N|k} - x_f^{i(k)})^T \tilde{Q}(\tilde{x}_{k+N|k} - x_f^{i(k)}) + \\ + \sum_{l=0}^{N-1} (\tilde{u}_{k+l|k}^T - u_f^{i(k)}) \tilde{R}(\tilde{u}_{k+l|k} - u_f^{i(k)}), \quad (18)$$

with $(x_f^{i(k)}, u_f^{i(k)})$ represents the optimal target positions and the associated control laws at current time k, $i = 1, \ldots, N_a$. Here $\tilde{Q} = \tilde{Q}^T \geq 0$, $\tilde{R} > 0$ are the weighting matrices with appropriate dimensions, $\tilde{P} = \tilde{P}^T \geq 0$ defines the terminal cost and N denotes the length of the prediction horizon.

First let us summarize in the following algorithm the receding horizon strategy together with task assignment mechanism:

Algorithm 1. Centralized scheme strategy for a group of agents

```
   Input: initial positions of the agents
1  compute the safety regions associated to each agent;
2  compute the minimal configuration as in (11);
3  compute the matrices defining the cost function (18);
4  for k = 1 : k_max do
5      if the safety regions are identical then
6          execute task assignment x^i_{k_ref} → x^i_f as in (13);
7      end
8      find the optimal control action u* as in (17);
9      compute the next value of the state: x̃_{k+1} = A_g x̃_k + B_g ũ_k
10 end
```

Due to the fact that we use invariant sets, steps 1, 2 and 3 can be executed in an off-line procedure. In the on-line part of the algorithm, we apply a finite horizon trajectory optimization: in step 6 we execute a task assignment if possible (only if the safety regions are identical) and then proceed with the actual computation of the receding horizon control (step 8). Finally, the first component of the resulting control sequence is effectively applied to the global system (step 9) and the optimization procedure is reiterated using the available measurements based on the receding horizon principle [9].

Remark 5. Although functional, this scheme will not scale favorably with an increased number of agents or a large prediction horizon due to the numerical difficulties. In particular, the mixed programming algorithms are very sensitive to the number of binary auxiliary variables. In this case a decentralized approach is to be envisaged in order to minimize the numerical computations.

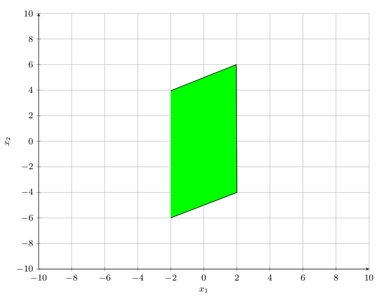
Remark 6. Note that although desirable, an increase in the length of the prediction horizon is not always practical, especially when using mixed-integer programming. We observed that a two-stage MPC, where in the first stage a task assignment procedure is carried and in the second, the usual optimization problem is solved offers good performances with a reduced computational effort.

5 Illustrative Example

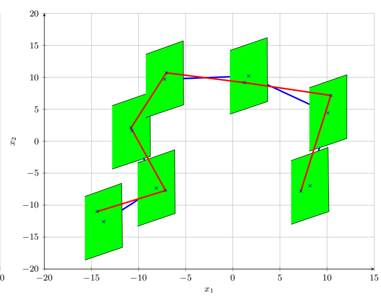
For the illustrative example we consider that each of the agent is described by the following dynamics and disturbances:

$$A_i = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\mu_i}{m_i} & 0 \\ 0 & 0 & 0 & -\frac{\mu_i}{m_i} \end{bmatrix}, \; B_i = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m_i} & 0 \\ 0 & \frac{1}{m_i} \end{bmatrix}, \; \mathcal{W}^i = \left\{ w^i : |w^i| \leq \begin{bmatrix} 0.5 \\ 0.3 \\ 0.5 \\ 0.2 \end{bmatrix} \right\}, \tag{19}$$

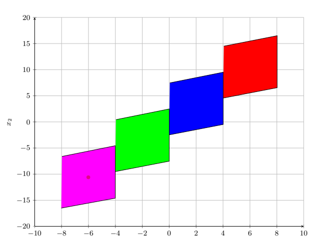
where $[x^i \; y^i \; v_x^i \; v_y^i]^T$, $[u_x^i \; u_y^i]^T$ are the state and the input of each system. The components of the state are the position (x^i, y^i) and the velocity (v_x^i, v_y^i) of the i^{th} agent, $i = 1, \ldots, N_a$. The parameters m_i, μ_i are the mass and the damping factor, respectively.



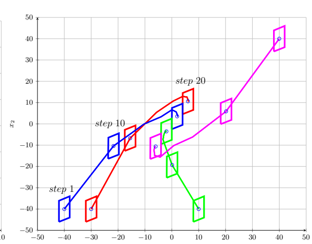
(a) Projection of the RPI set on the position subspace



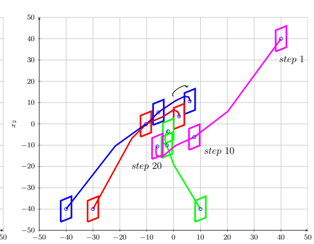
(b) The trajectories of the "real" and "nominal" system



(c) The minimal configuration of four homogeneous agents



(d) The evolution of 4 homogeneous agents with task assignment



(e) The evolution of 4 homogeneous agents withought task assignment

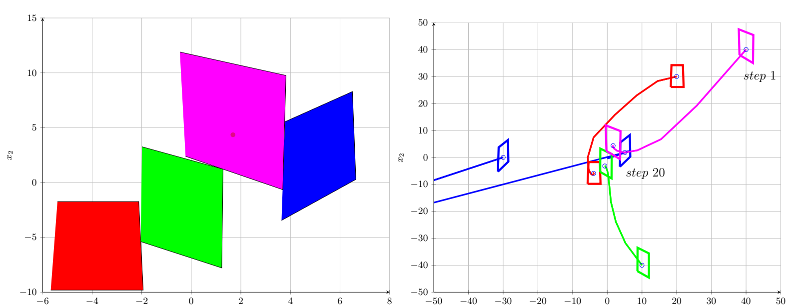
Fig. 1. The tight formation of 4 homogeneous agents

First, let us consider $N_a = 4$ homogeneous agents with $m_i = 45$kg, $\mu_i = 15$Ns/m, $i = 1, \ldots, 4$. For the sake of illustration, we construct the RPI sets (i.e.,the safety regions) for the homogeneous agents affected by disturbances. Using pole placement methods we derive the feedback gain matrices

K_i, $i = 1, \ldots, 4$, which placed the poles of the closed-loop system in the 0.6 to 0.9 interval. The RPI sets S_i are obtained as detailed in [13] and the projection on the position subspace of one set is depicted in Figure 1 (a). For this system a nominal trajectory (2) (in blue) is constructed and one observes in Figure 1 (b) that any trajectory of system (1) affected by disturbance will verify relation (5) (i.e., resides in a tube described by the RPI set S_i). Furthermore, by solving the mixed-integer optimization problem (11) we obtain the set of target positions $T = \left\{(x_f^i, u_f^i)\right\}, i = 1, \ldots, 4$ as in (12), which satisfy the anti-collision conditions (9) (see, Figure 1 (c)). As it can be seen in Figure 1 (c), for the case of agents with the same dynamics (as detailed also in Remark 4) the equilibrium points of (12) stay on the same subspace, which in this particular case is a line passing through the origin.

We next apply the receding horizon scheme (17) for the global system with a prediction horizon $N = 2$ and the tunning parameters $\tilde{P} = 5I_4$, $\tilde{Q} = I_4$, $\tilde{R} = I_2$. The optimal trajectories for the agents are obtained such that the set of target points is reached through task optimization and under state constraints. We summarized the details in Algorithm 1. The effectiveness of the present algorithm is confirmed by the simulation depicted in Figure 1 (d), where the evolution of the agents is represented at three different time instances. The agents successfully reach their target positions in the predefined formation without violating the constraints and with a minimum cost.

For comparison purposes we execute Algorithm 1 with and without the task assignment stage. As it can be seen in Figure 1 (e) for a prediction horizon of $N = 2$ and without the task assignment procedure, the agents do not converge to the desired configuration (two agents, depicted in blue and red, respectively switch places). Note that if the prediction horizon is long enough (in this particular case $N = 8$) the desired configuration is achieved but the computational complexity of the mixed-integer optimization problem (11) increases significantly.



(a) The target configuration of 4 heterogeneous agents (b) The evolution of 4 heterogeneous agents

Fig. 2. The tight formation of 4 heterogeneous agents

Second, let us consider $N_a = 4$ heterogeneous agents with different values of the parameters m_i and μ_i, $i = 1, \ldots, 4$. The prediction horizon is $N = 7$ and the tunning parameters are $\tilde{P} = 50I_4$, $\tilde{Q} = I_4$, $\tilde{R} = I_2$. Following the same procedure we depict in Figure 2 (a) the agents with the associated safety regions in a minimal configuration. Furthermore, in Figure 2 (b) we illustrate the evolution of the agents at two different time instances.

6 Conclusions

In this paper, we first present several tools in order to provide a systematic off-line procedure for the control of a group of agents towards a minimal configuration. Second, in real-time a two stage receding horizon control design is adopted for driving the agents to the predefined formation. Also, we provide several remarks, leading to computational improvements of the mixed-integer techniques used to assure a collision free behavior along the evolution of the agents. The results are presented through some illustrative simulations of several examples. The current research is to develop software-in-the-loop simulations and subsequent flight tests for the control of small Unmanned Aerial Vehicles (UAVs). The vehicle dynamics is simulated by a Piccolo software and then, the control algorithm is transmitted in real flight simulations through a communication routine running on a PC on the ground.

References

1. Balch, T., Arkin, R.: Behavior-based formation control for multirobot teams. IEEE Transactions on Robotics and Automation 14(6), 926–939 (1998)
2. Girard, A., de Sousa, J., Hedrick, J.: Coordinated Control of Agent Formations in Uncertain, Dynamic Environments. In: Proceedings of the European Control Conference, Porto, Portugal (2001)
3. Grundel, D., Murphey, R., P.M., P.: Cooperative systems, Control and optimization, vol. 588. Springer (2007)
4. Hallefjord Kurt, O.: Solving large scale generalized assignment problems–An aggregation/disaggregation approach. European Journal of Operational Research 64(1), 103–114 (1993)
5. Jadbabaie, A., Lin, J., Morse, A.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Transactions on Automatic Control 48(6), 988–1001 (2003)
6. Jünger, M., Liebling, T., Naddef, D., Nemhauser, G., Pulleyblank, W.: 50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art. Springer (2009)
7. Lafferriere, G., Caughman, J., Williams, A.: Graph theoretic methods in the stability of vehicle formations. In: Proceedings of the 2004 American Control Conference, vol. 4, pp. 3729–3734 (2005)
8. Loechner, V., Wilde, D.: Parameterized polyhedra and their vertices. International Journal of Parallel Programming 25(6), 525–549 (1997)
9. Mayne, D., Rawlings, J., Rao, C., Scokaert, P.O.: Constrained model predictive control: Stability and optimality. Automatica 36, 789–814 (2000)

10. Mayne, D., Seron, M., Raković, S.: Robust model predictive control of constrained linear systems with bounded disturbances. Automatica 41(2), 219–224 (2005)
11. Palekar, U., Karwan, M., Zionts, S.: A branch-and-bound method for the fixed charge transportation problem. Management Science 36(9), 1092–1105 (1990)
12. Prodan, I., Olaru, S., Stoica, C., Niculescu, S.-I.: Predictive control for tight group formation of multi-agent systems. In: Proceedings of the 18th IFAC World Congress, Milano, Italy, pp. 138–143 (2011)
13. Raković, S., Kerrigan, E., Kouramas, K., Mayne, D.: Invariant approximations of the minimal robust positively invariant set. IEEE Transactions on Automatic Control 50(3), 406–410 (2005)
14. Richards, A., How, J.: Aircraft trajectory planning with collision avoidance using mixed integer linear programming. In: Proceedings of the 21th American Control Conference, Anchorage, Alaska, USA, pp. 1936–1941 (2002)
15. Stoican, F., Prodan, I., Olaru, S.: On the hyperplanes arrangements in mixed-integer techniques. In: Proceedings of the 30th American Control Conference, San Francisco, CA, USA, pp. 1898–1903 (2011)
16. Tanner, H., Jadbabaie, A., Pappas, G.: Flocking in fixed and switching networks. IEEE Transactions on Automatic Control 52(5), 863–868 (2007)

Dynamic Customization of a Remote Conversation Support System: Agent-Based Approach

Masaya Morita[1] and Kazuhiro Kuwabara[2]

[1] Graduate School of Science and Engineering, Ritsumeikan University
[2] College of Information Science and Engineering, Ritsumeikan University
1-1-1 Noji-Higashi, Kusatsu, Shiga 525-8577 Japan

Abstract. This paper proposes an agent-based approach to customizing a remote conversation support system for cognitively handicapped people. This remote conversation support system is designed as a web application to assist with conversation over the Internet. The system is built from 'gadgets,' each of which implements a particular conversation support function. Since the need for conversation support varies from person to person, such a system must be customized to suit the requirements of the two users who conduct the conversation. The proposed approach introduces a user agent that corresponds to a human user. The negotiation protocol for selecting the necessary gadgets is defined based on the FIPA interaction protocol. This paper describes how the proposed protocol can be used to determine the gadgets to be used during run-time.

Keywords: interaction protocol, gadget, conversation support.

1 Introduction

This paper proposes an agent-based approach to dynamically customizing a remote conversation support system. We are developing a conversation support system over the Internet, mainly targeted to people with cognitive handicaps such as aphasia [4]. This system is implemented as a web application that provides various types of support for conversation between a handicapped person and a conversation partner over the network. For example, the system allows access to a topic database containing various topic keywords to facilitate conversation. The system also provides a remote co-browsing function so that users at remote sites can see the same browser screen. This function is utilized to implement a set of 'tools' to help the person with a cognitive handicap indicate his or her intention.

Given the variety of handicaps affecting communication, the tools needed to indicate one's intentions and to facilitate conversation will vary from person to person. It is possible to make all of the tools easily accessible from the computer screen, but this tends to make the system overly complicated. On the other hand,

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 566–575, 2012.

it is difficult for the targeted user to select the necessary functions himself or herself, especially a person who is not familiar with computers. Furthermore, since the system is intended for remote conversation support, each user can customize the system only after giving consideration to the needs of the user at the other end.

To solve this problem, this paper proposes a 'user agent' for each human user and a means to let the user agents negotiate with each other to decide on the functions to be used. By giving a user agent information on its corresponding human user's preferences, it becomes possible for the user agents to dynamically configure a system that suits the needs of both human users. This would be particularly beneficial for users who are not accustomed to using computers.

This paper is structured as follows. The next section discusses related works. Section 3 outlines the remote conversation support system for which the proposed mechanism is designed, and Section 4 describes an agent-based negotiation mechanism for the customization function. Section 5 shows an example usage of the proposed mechanism, and the final section concludes this paper.

2 Related Works

Gadgets are often used in personalizing the user interface, especially for web-based applications. Users can select their favorite gadgets to suit their own needs without programming.

In order to automatically adapt the user interface, an approach using ontology has been proposed [6]. By introducing adaptation rules, a system can adapt web pages. This has been applied to the construction of an e-Government web portal. This work has been adopted for customizing a rich Internet application, but its use in collaborative applications has not been addressed.

For applications involving multiple users, the concept of distributed interface orchestration has been proposed and applied to the construction of a collaborative search application [2]. In this approach, a work-flow is used to describe a task that involves multiple users. The work focuses on the work-flow but not on the kind of real-time collaboration this paper addresses.

The EzWeb architecture has been proposed for application to enterprise collaboration [7]. In this approach, the users use off-the-shelf gadgets to assemble a new service. The target users are so-called knowledge workers, and this work emphasizes bringing about innovation through co-creating a new service.

As an agent-based approach, the Isatine framework has been proposed [5]. In this framework, the user interface is adapted by the collaboration of multiple agents. The adaptation is performed based on the task model of the target application, and various hardware and software sensors are applied to recognize the user context. Each agent is assigned a particular role, such as context detection or handling the adaption process. This framework is targeted mainly to a user interface for a particular task performed by a single user, and thus it is not readily applicable to customizing the user interface of a communication support system.

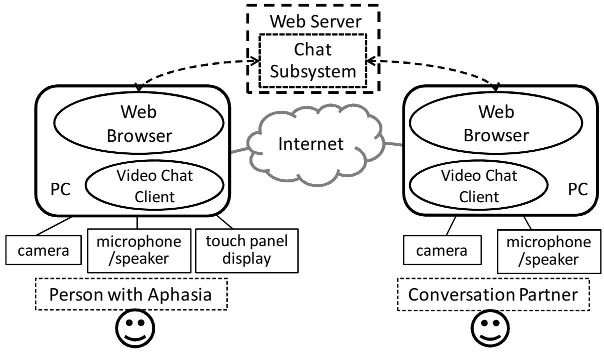


Fig. 1. Overview of remote conversation support system

3 Remote Conversation Support System

3.1 Overview

The proposed remote conversation support system is implemented as a web application (Figure 1). The core of the system is a real-time chat system that runs over a browser. A user (a person with a cognitive handicap or his/her conversation partner) uses the browser to access the contents of the web server. The system uses a 'server push' technique to allow the users at both ends to see the same web browser contents. The video chat (in the current implementation, we use 'Skype') is used for audio and video communication in addition to sharing the browser screens. For the person with a cognitive handicap, a touch panel display is assumed to be used so that the user can easily show his or her intention by touching a certain area on the screen.

The system includes various 'tools' to facilitate communication, such as a 'yes/no' tool and a calendar tool. These tools are designed to satisfy the various needs of people with cognitive handicaps such as aphasia. For example, the calendar tool is intended to help the user answer a question about a certain date. If the user cannot speak up the date, he or she can touch the date in the calendar shown on the screen as part of the 'calendar' tool. In the calendar tool, the same calendar is displayed on both browsers, and the date selected (by touch) is highlighted on both users' screens. Other tools include a 'scale' tool for answering a question about the degree of something (for example, "how much do you like sports?") by moving a slider bar shown on the browser screen. The movement of the slider bar is synchronized between the two browsers.

In addition to these tools, the remote conversation support system implements a function to access various web services such as a web search engine, a web image album for sharing photos, and the topic database created from the vocabulary data file [9].

These functions can be selected by clicking (touching) a corresponding tab on the screen. Figure 2 is a screenshot of part of this system, with three tabs placed

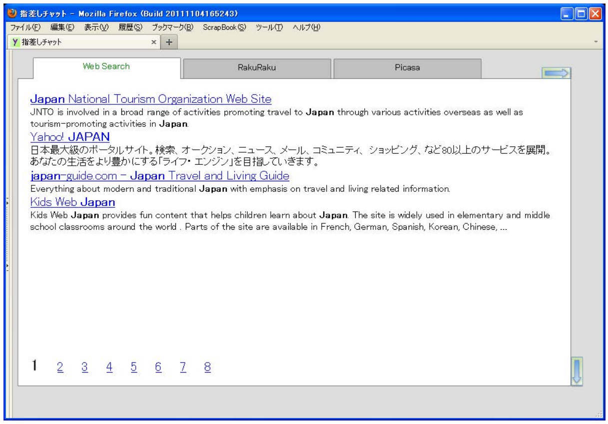


Fig. 2. Screenshot of conversation support system (partial)

in the upper part of the screen. In this example, the opened tab shows results of a web search, with other tabs giving access to the topic database ('RakuRaku') and a web image album service ('Picasa').

3.2 Gadget-Based Customization

In order to make it easier to customize the system, a gadget-based architecture is introduced. Each gadget is designed to implement a certain function. For example, the functions mentioned above, such as web-search and web photo album, are provided as gadgets. By selecting appropriate gadgets, the system can be customized without programming. However, since people with cognitive handicaps tend to be unfamiliar with computers, it is not practical to provide all of the functions and let them select the needed functions during run-time. Instead, it is preferable to automatically select and provide the appropriate functions (gadgets) for them.

Since remote conversation support involves multiple parties, compatible gadgets need to be selected at both ends. When one party prefers a certain gadget, the other party needs to agree on the selection of that gadget. Otherwise, the communication support would break down. Consequently, it is necessary to reach an agreement on the selection of gadgets at both ends.

4 Agent-Based Approach

In order to facilitate the negotiation between the two ends, we introduce an agent-based approach. For each user of the system, we provide a corresponding user agent. The user agent has information on its corresponding human user,

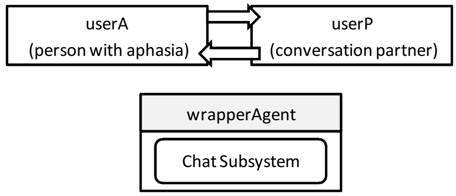


Fig. 3. Agents in the system

such as his or her preferences. When a conversation session starts, the user agents negotiate over which gadgets to adopt for the human users.

4.1 Types of Agents

When a user logs into the system, the corresponding user agent is loaded in the web server portion of the system. We assume a simple agent monitor that handles the communication between the user agents. The agent monitor is run at the web server. In order to treat the remote conversation support itself as an agent, a wrapper agent is designed (Figure 3).

We modeled the agent communication following the FIPA agent specifications [3]. We used several communicative acts and introduced interaction protocols based on those defined by FIPA.

4.2 Interaction Protocol for Gadget Selection

We defined an interaction protocol for the two user agents to apply in negotiating over which gadgets to use. This protocol, called 'mutual-propose-protocol,' is defined based on the FIPA propose interaction protocol [3], and it introduces an additional `counter-propose` message (Figure 4). In this protocol, one user agent acts as an initiator and starts the negotiation. First the initiator agent sends a `propose` message to the other user agent to propose a gadget to use. The agent that receives this message replies with an `accept-proposal` message, `reject-proposal` message, or `counter-propose` message. By allowing a `counter-propose` message, the non-initiator agent can make a proposal in the negotiation process. The agent that receives a `counter-propose` message replies with an `accept-proposal`, `reject-proposal`, or another `counter-propose` message. When the `counter-propose` message is sent again, the behavior of this interaction protocol effectively iterates.

4.3 Forming Proposals

The interaction protocol defined above only concerns the interaction between the user agents. In order to use this protocol, the user agents need to decide on

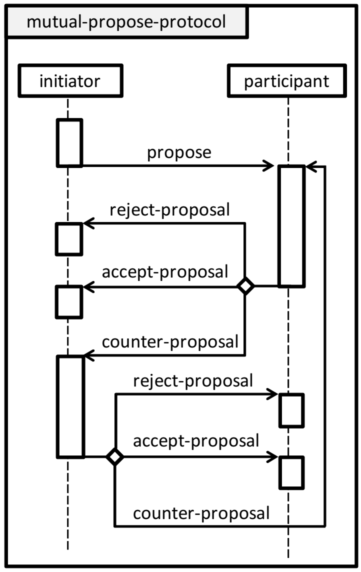


Fig. 4. Mutual propose protocol

the contents of the gadget proposal. The following two types of information are used.

- Usage history of the gadgets
  The history of the system's usage is recorded. In addition to the number of sessions, the gadget that was used is recorded. The history is represented using the resource description framework (RDF) [8].
- Tags attached to the gadgets
  When the user does not have a history of using the system, the user agent selects a candidate gadget based on the user's preferences. The user preferences are represented as keywords. A gadget is also assumed to have tags that contains keywords. By examining the tags' information, a gadget is selected.

Figure 5 shows an example of the user's information that includes a usage history of the gadgets in the RDF format. The user's preference is also represented by tags. The usage history contains gadgets used and their usage time.

5 Examples of Agent Negotiation

Let us consider an example where a person with aphasia (`userA`) and a conversation partner (`userP`) conduct a conversation. The chat subsystem is wrapped as an agent (`wrapperAgent`). The negotiation starts with a `request` message sent from the `wrapperAgent`. The interaction protocol between the `wrapperAgent` and the user agents uses a simplified version of the FIPA request protocol.

```
@prefix sem: <http://semlab.jp/ychat#> .

<http://semlab.jp/ychat/users/userA>
  sem:userType sem:aphasia ;
  sem:tags [ sem:tag "game", "rehabilitation" ] ;
  sem:historyOfConversation
    [ sem:gadgetUsed <http://semlab.jp/ychat/gadgets/Web_Search>
      sem:gadgetUseTime " 570.22 " ;
      sem:numberOfTimes "2" ; ] ,
    [ sem:gadgetUsed <http://semlab.jp/ychat/gadgets/Topic_DB>
      sem:gadgetUseTime " 323.53 " ;
      sem:numberOfTimes "2" ; ] .
```

Fig. 5. User's information (partial)

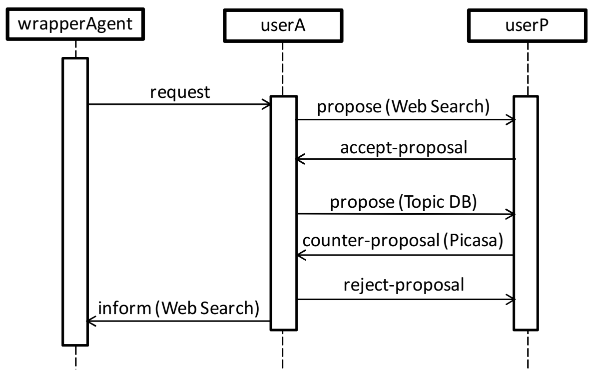


Fig. 6. Message exchange in the example

The message format is based on the FIPA agent communication language. Since the usage history and gadget information are represented using RDF, the content language of the agent message is also based on RDF.

5.1 Example of Selecting a Gadget Based on the Usage History

In this example, the chat system is assumed to have three types of gadgets (Web-search, Topic-DB, and Picasa (Web photo album)). The example's message sequence is shown in Figure 6.

Let us assume that agent `userA` receives a `request` message from `wrapperAgent`. After receiving this message, `userA` calculates a candidate gadget to use. Let us suppose the Web-search gadget and the Topic-DB gadget are retrieved from the usage history. `userA` initiates the negotiation by sending a `propose` message (Figure 7) to propose the usage of the Web-search gadget.

Agent `userP` also calculates a list of gadget candidates, and let us suppose that the Web-search gadget and the Picasa (photo album) gadget are selected. When `userP` receives a `propose` message with the content of the Web-search gadget, it first checks whether the proposed gadget is included in its own candidate list.

```
(PROPOSE
 :sender (agent-identifier :name userA)
 :receiver (set (agent-identifier :name userP))
 :reply-to (set (agent-identifier :name userA))
 :content
 "@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
  @prefix sem: <http://semlab.jp/ychat#> .
  [] a sem:ACL_CONTENT ;
     sem:act "NegotiationUseGadget" ;
     sem:argument
       [ sem:gadget <http://semlab.jp/ychat/gadgets/Web_Search> ] ."
 :language rdf  :protocol mutual-propose
 :conversation-id userA1323641239640)
```

Fig. 7. `propose` message example

```
(ACCEPT-PROPOSAL
 :sender (agent-identifier :name userT)
 :receiver (set (agent-identifier :name userA))
 :content
 "@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
  @prefix sem: <http://semlab.jp/ychat#> .
  [] a sem:ACL_CONTENT ;
     sem:act "NegotiationUseGadget" ;
     sem:argument
       [ sem:gadget <http://semlab.jp/ychat/gadgets/Web_Search> ] ."
 :reply-with  userA1323641406578  :language  rdf
 :protocol  mutual-propose
 :conversation-id  userA1323641239640)
```

Fig. 8. `accept-proposal` message example

Since it is included in this example, `userP` replies with an `accept-proposal` message (Figure 8).

After `userA` receives the `accept-proposal` message, it tries to propose another gadget, namely the Topic-DB gadget with a `propose` message. Since this gadget is not included in the candidate gadget list of `userP` but another gadget, namely the Picasa (photo album) gadget, is in this list, `userP` sends a `counter-propose` message with this gadget in its contents as a reply. Since the proposed Picasa gadget is not in the candidate list of `userA`, it sends a `reject-proposal` message. `userA` then sends to the `wrapperAgent` a list of gadgets that includes the Web-search gadget as the gadget to use.

5.2 Example of Selecting a Gadget Based on Gadget's Tags

Let us now consider an example that uses the tags attached to a gadget. In this example, the chat system has two gadgets: an Othello game gadget and a Word game gadget. The tag of 'game' is attached to these gadgets (Figure 9). When the negotiation phase starts, agent `userA` sends a `propose` message to agent `userP` to use the Othello gadget. If `userP` has the information that its human user is not fond of the game of Othello, it considers sending a `reject-proposal` message. However, simply sending a `reject-proposal` message will not result in

```
@prefix sem: <http://semlab.jp/ychat#> .

<http://semlab.jp/ychat/gadgets/Othello>
  sem:gadgetName "Othello" ;
  sem:gadgetTags
    [ sem:gadgetTag "aphasia" , "therapist" , "game" ] .

<http://semlab.jp/ychat/gadgets/Word_game>
  sem:gadgetName "Word game" ;
  sem:gadgetTags
    [ sem:gadgetTag "aphasia" , "therapist" , "game", "rehabilitation" ] .
```

Fig. 9. Gadget information

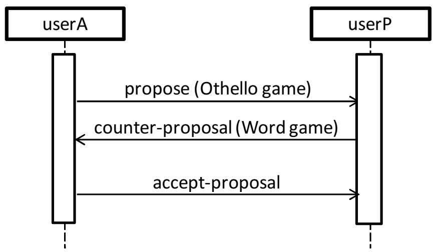


Fig. 10. Another example of message exchange

an agreement over gadgets, so `userP` tries to make a counter proposal instead. In this case, since the Word game gadget has the same tag of 'game,' `userP` sends a `counter-propose` message that contains the Word game gadget. If `userA` agrees to using the Word game gadget, the negotiation ends (Figure 10).

Through the process outlined above, the user agents `userA` and `userP` exchange information and decide on the gadget to use.

6 Conclusion

This paper proposed an agent-based approach to gadget selection in a remote conversation support system for people with cognitive handicaps such as aphasia. In a collaborative system such as a conversation support system, customization cannot be performed by one party alone. The users who participate in the conversation need to agree on the functions they use. In this system, a user agent that has information on its corresponding human user can negotiate with the other user agent to decide on the gadgets to be used.

Currently, we are implementing the proposed mechanism in an existing remote conversation support system. There remain various issues that require further work. For example, how to determine a set of gadgets for negotiation, especially when a new gadget is introduced, needs to be clarified. Effective use of ontology would be one possible approach to tackling this problem. Even without the history of previous usage, a gadget could be selected as a candidate for use. In order to achieve this capability, an ontology for describing a gadget needs to be developed.

In this paper, we focused on supporting a conversation between two people and described the negotiation mechanism between two user agents for agreeing upon the gadgets to use. Since group video chat involving more than two people is now available, the proposed approach will be extended to handle a group negotiation among more than two users.

In addition, from the viewpoint of the conversation support, we can consider extending the role of user agents so that the agents can be directly involved in the conversation. We plan to examine a way to integrate an approach of conversational agents (such as [1]) into the proposed system.

References

1. Crockett, K., O'Shea, J., Bandar, Z.: Goal Orientated Conversational Agents: Applications to Benefit Society. In: O'Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS, vol. 6682, pp. 16–25. Springer, Heidelberg (2011)
2. Daniel, F., Soi, S., Casati, F.: Distributed User Interface Orchestration: On the Composition of Multi-User (Search) Applications. In: Ceri, S., Brambilla, M. (eds.) Search Computing II. LNCS, vol. 6585, pp. 182–191. Springer, Heidelberg (2011)
3. Foundation for Intelligent Physical Agents: FIPA specifications, http://www.fipa.org/
4. Kuwabara, K., Shimode, Y., Miyamoto, S.: Agent-Based Remote Conversation Support for People with Aphasia. In: Jędrzejowicz, P., Nguyen, N.T., Howlet, R.J., Jain, L.C. (eds.) KES-AMSTA 2010, Part I. LNCS, vol. 6070, pp. 371–380. Springer, Heidelberg (2010)
5. López-Jaquero, V., Vanderdonckt, J., Montero, F., González, P.: Towards an Extended Model of User Interface Adaptation: The Isatine Framework. In: Gulliksen, J., Harning, M., Palanque, P., van der Veer, G., Wesson, J. (eds.) EIS 2007. LNCS, vol. 4940, pp. 374–392. Springer, Heidelberg (2008)
6. Schmidt, K.-U., Dörflinger, J., Rahmani, T., Sahbi, M., Stojanovic, L., Thomas, S.M.: An User Interface Adaptation Architecture for Rich Internet Applications. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 736–750. Springer, Heidelberg (2008)
7. Soriano, J., Lizcano, D., Ca Nas, M.Á., Reyes, M., Hierro, J.J.: Fostering innovation in a mashup-oriented enterprise 2.0 collaboration environment. System and Information Science Notes 1, 62–69 (2007)
8. W3C: Resource Description Framework (RDF), http://www.w3.org/RDF/
9. Yasuda, K., Nemoto, T., Takenaka, K., Mitachi, M., Kuwabara, K.: Effectiveness of a vocabulary data file, encyclopaedia, and Internet homepages in a conversation-support system for people with moderate-to-severe aphasia. Aphasiology 21(9), 867–882 (2007)

Developing Agile Supply Chains – System Model, Algorithms, Applications

Katarzyna Grzybowska[1] and Gábor Kovács[2]

[1] Poznan University of Technology, Faculty of Engineering Management, Chair of Production Engineering and Logistics, Strzelecka 11, 60-965 Poznan, Poland
katarzyna.grzybowska@put.poznan.pl
[2] Budapest University of Technology and Economics, Faculty of Transportation Engineering and Vehicle Engineering, Department of Transportation Technology, Műegyetem rkp 3, 1111 Budapest, Hungary
kovacsg@kku.bme.hu

Abstract. Nowadays, a modern company operates as a network of companies, not as a single company. These companies are called virtual enterprises, whose main purpose is the maximal fulfilment of costumer needs. Based on this, the paper presents the construction of agile supply chains and as an example it details the model, the algorithms and the applications of a modern electronic freight and warehouse exchange, which is a type of virtual enterprises.

Keywords: agile, supply chain, e-commerce, optimization, decision supporting.

1 Introduction

The interest in the concept of Supply Chain (SC) results from the engagement of theoreticians and practical researchers in the integrated flow of goods from the supplier to the end client on an increasingly turbulent, unpredictable market. It is in the 1980s that organizational solutions and the general concept of Supply Chain Management (SCM) became an alternative to the traditional (usually transactional) manner in which the relationship is perceived and the manner in which a business operates between cooperating suppliers and recipients. The SC is a network of organizations that are involved in different processes and activities that produce value in the form of products and services in the hands of the ultimate consumer [1]. The Supply Chain is a metastructure [2].

There can be distinguished constitutive elements (characteristics) of the supply chains. These constitutive elements allowing for the identification of the supply chains that differ considerably [3]: (1) supply chain size, (2) the ascribed roles of supply chain participants, (3) status of participants of a supply chain, (4) coherence of the supply chain, (5) communication in a supply chain, (6) interactions within a supply chain.

The Supply Chains are longer, more dynamic and riskier. They are much more complex. A metastructure is characterized by a dynamic holarchy of cooperating holons (commercial entities) and holonic system (HS).

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 576–585, 2012.

The greater the supply chain grows, the less coherent and lacking close relationships are. In turn, this results in a situation in which in such a metastructure's connections and dependencies may vary in permanence. One differentiates permanent links (the so-called core supply chain) and dynamic ones which change depending on the task carried out (the so-called temporary links). After cooperation is concluded, the temporary links become disconnected from the supply chain cooperation. The SC is a concept designed to manage entire supply chains consisting of numerous participating organizations [4].

SCM is based on the understanding that the challenges of the present-day competition cannot be effectively faced through isolated changes in the individual companies but through the development of collaboration between the participants in the production and the delivery of products from the initial sources to the end customers [5].

New market and economy need the Agile Supply Chain (ASC). The ASC must possess several distinguishing characteristics as Fig. 1 suggests [6]: the ASC (1) is market sensitive; (2) uses information technology to share data between buyers and suppliers, in effect, creating a virtual supply chain; (3) is integrated – the share of information between supply chain partners (collaborative working between buyers and suppliers, joint product development, common systems and shared information; "extended enterprise" as it is often called, there can be no boundaries and an ethos of trust and commitment must prevail); (4) is a confederation of partners linked together as a network.

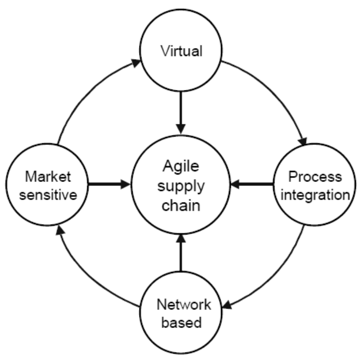


Fig. 1. The Agile Supply Chain [9]

The techniques of information sharing (i.e., information centralization) can be supported by information technologies such as e-Hubs [7]. The basis of these information technologies is currently the Internet and the application of multiagent systems in SC. Intelligent agents are a new paradigm of software system development. Nowadays, it seems that researchers agree on the following definition proposed by Wooldridge and Jennings [8]. The term "agent" denotes a hardware or (more usually) software-based computer system, that has the following characteristics: (1) Autonomy: agents operate without the direct intervention of humans or others, and has some kind of control over its actions and internal state; (2) Social ability: agents interact with other agents (and possibly humans) via some kind of agent-communication language; (3) Reactivity: agents perceive their environment, (which may be the physical world, an user, a collection of other agents, the Internet, or perhaps all of these combined), and respond in a timely fashion to changes that occur in it; (4) Pro-activeness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behaviour by taking the initiative [9].

Agility in the Supply Chain is the ability of the supply chain as a whole and its members to align rapidly the network and its operations to the dynamic and turbulent requirements of the customers. The main focus is on running business in network structures with an adequate level of agility to respond to changes as well as proactively anticipate changes and seek new emerging opportunities [10]. They should be able to be flexible, responsive and adaptable to changing market conditions. This can be achieved through collaborative relationship, process integration, information integration, and customer/marketing sensitivity achieving customer satisfied objectives.

The SC is agile and responsive in an increasingly competitive and fast-paced business world, when has informational nervous systems that extend beyond their own corporate boundaries (Fig. 2).

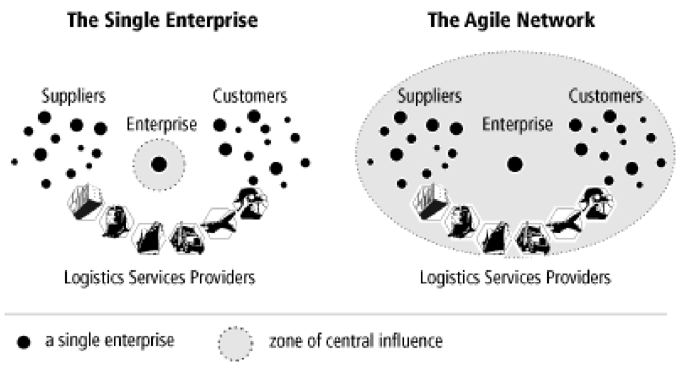


Fig. 2. Informational nervous systems in Single Enterprise and Agile Network [11]

2 Developing Agile Supply Chains - Four Stages of Supply Chain Integration According to Stevens

Stevens identified four stages of SC integration and discussed the planning and operating implications of each stage [14]: (Stage 1) Represents the base line case. The SC is a function of fragmented operations within the individual company and is characterized by staged inventories, independent and incompatible control systems and procedures, and functional segregation; (Stage 2) Begins to focus internal integration, characterized by an emphasis on cost reduction rather than performance improvement, buffer inventory, initial evaluations of internal trade-offs, and reactive customer service; (Stage 3) Reaches toward internal corporate integration and characterized by full visibility of purchasing through distribution, medium-term planning, tactical rather than strategic focus, emphasis on efficiency, extended use of electronics support for linkages, and a continued reactive approach to customers; (Stage 4) Achieves SC integration by extending the scope of integration outside the company to embrace suppliers and customers.

3 Developing Agile Supply Chains - Three Stages of ASC

The development of the SC takes place in three main stages, in which a total of seven phases may be differentiated [15]:

- Stage 1: Strengthening the contact
- Stage 2: Maintaining the contact
- Stage 3: Loosening the contact

The first step in strengthening an enterprise's business contacts is the initiation of the contact. In this phase, the selection of the business partner takes place along with the initial assessment of the possible terms and conditions of cooperation. Coordination mechanisms are also started up. The performance of a supply chain depends critically on how its members coordinate their decisions. Sharing information is the most basic form of coordination in Supply Chains.

Further positive business relations lead to integration among enterprises. To a large extent, this phase is based on the creation of a certain cooperative of interests. In the contract maintenance phase – the integration phase – the following may be observed: (1) increased communication; (2) greater readiness to cooperate; (3) an increased plan for the actions of others; (4) the more effective achievement of the intended objectives; (5) an increased level of satisfaction in cooperation.

In this integrated phase, the enterprises also expect more from each other. The integration phase ends if the enterprises do not overcome the crisis periods. The limiting business contacts phase begins. The result of the extending limitation of the business relations is the impasse.

4 Agile Supply Chains – Model of Electronic Freight and Warehouse Exchanges

With the help of the Internet, information can be sent to participants of business processes in the fraction of a second, which, by accelerating and optimizing these processes, facilitates an easy overview and comparison of supply and demand [16], [17].

For this reason, electronic marketplaces have emerged in numerous fields, such as freight exchanges in the field of carrier services. Freight exchanges create a meeting point for freighters and consigners. Consigners can advertise their freight tasks for shipment in the catalogue of the marketplace; similarly, freighters can make their bid for cargo holds. Moreover, the users of these exchanges can choose the most suitable offer by using different search algorithms. Warehouse exchanges sell free warehouse space/task either with a simple advertisement or by using search functions.

Nevertheless, because underdeveloped transactional solutions set back the development of such exchange types. The overall structure of freight and warehouse exchanges is supplemented by showing other modules and services, too, in the forthcoming subsections. These could be important and useful modules in the future.

The developed freight and warehouse exchange offers the following main services [18]: e-commerce toolbar; multi-criteria decision supporting algorithms (choose the best offer); optimization algorithms (optimize the logistics processes); other functions (e.g. statistics, blacklists, data maintenance, etc.). The aims of the electronic freight and warehouse exchange: to advertise freight/storage capacities/tasks; to choose suitable offers based on e-commerce methods and complex optimum criteria [19]; to support complex logistic processes (e.g. combined transport, city logistics, etc.).

When a new customer who has not used the system before wants to register, their personal data and contact details must be entered. Also, we can specify here the filtering criteria which are necessary for the automatic offer sending, and we can also add our personal negative experiences.

Consigners specify the details of their freight/storage tasks (e.g. temporal/spatial/physical parameters, etc.). Logistics providers can do a search based on the mentioned parameters. It is also possible to find backhaul and to look at the whole task offer here, too. Logistics providers can offer their freight/storage capacities by displaying all relevant information (e.g. temporal/spatial/physical parameters, etc.). Consigners can do a search based on the mentioned parameters. We can take a look at the whole capacities offer here, too.

After giving our personal data, the system allows us to enter filtering criteria which will help us to choose quickly from the latest offers (automatic offer sending.). Such criteria are: time, spatial or physical limitations, etc.

There is opportunity to have a freight/storage commission through tender or auction [20], [21]. Experiences show that tenders for high-value, long-term and repetitive freight/storage tasks are worth advertising on the electronic freight and warehouse exchange.

5 Agile Supply Chains - Decision Supporting

One of the chief values of the developed electronic freight and warehouse system is the automatic application of multi-criteria evaluation methods that are well-known from books, but may not be used enough in practice. The developed mathematical method (MDA: Multi-criteria Decision supporting Algorithm) helps to evaluate tenders/auctions [22].

MDA is based on the principle of the AHP - Analytic Hierarchy Process - [23], [24], and other methods (e.g. caeteris paribus, mathematical maximal sensitivity analysis). MDA is an MS Excel and VBA-based (Visual Basic Application) application. By its decision making nature, it generates reports (Table 1.) that help making well-founded and agile decisions. The main goal:

$$\textbf{Weighted performance value => MAX!} \quad (1)$$

Table 1. MDA generated report in the course of a freight tender (example)

Main aspects		Sub aspects			Offers and their values					
Name	Weight	Name	Weight	Interpretation	1	2	3	4	5	Ideal
Fare (100 Euro)	0.4082	Fare (100 Euro)	1	lower	421	525	590	586	448	421
Deadline (day)	0.2041	Deadline (day)	1	lower	3	3	4	2	3	2
Proximity	0.1361	Proximity	1	higher	0.3457	0.3457	0.1728	0.0494	0.0864	0.3457
Services	0.102	Services	1	higher	0.125	0.25	0.125	0.25	0.25	0.25
Information connections	0.068	Information connections	1	higher	0.1509	0.2264	0.4528	0.0566	0.1132	0.4528
References	0.0816	General references	0.75	higher	0.2759	0.1379	0.1379	0.2759	0.1724	0.2759
Sensitivity		Trust	0.25	higher	0.1429	0.2857	0.2857	0.1429	0.1429	0.2857
Fare (100 Euro)	14.59%				1	0.8015	0.7127	0.7186	0.9393	1
Deadline (day)	4.93%				0.6667	0.6667	0.5	1	0.6667	1
Proximity	3.33%				1	1	0.5	0.1429	0.25	1
Services	4.48%				0.5	1	0.5	1	1	1
Information connections	4.35%				0.3333	0.5	1	0,125	0.25	1
References	0.53%				0.875	0.625	0.625	0.875	0.5938	0.875
			Offers		1	2	3	4	5	
			Weighted performance value		0.8254	0.863	0.631	0.6988	0.721	

Final order of offers						
Final order of offers	Offers	1	2	5	4	3
	Weighted performance value	0.8254	0.7863	0.721	0.6988	0.631

6 Agile Supply Chains - Optimization Algorithms and Applications: Agile Combined Transportation System

The basic function of electronic freight and warehouse exchanges is to establish connection between free freight and storage capacities and tasks. In the database of such online fairs there is high number of freight and storage capacity offers and tasks, which provides good optimization opportunity for logistics providers [25].

In the freight exchange the optimum search task may be formulated on the basis of the following objective function: those having free freight capacity wish to establish routes providing optimal profit from the freight tasks appearing in the freight exchange. Many freight tasks may be included into the route, but a new freight task may be commenced only after the completion of the previous one. The objective function is to reach the maximum profit.

In the warehouse exchange those having free storage capacity wish to choose several from the available storage tasks by setting the goal of ideal exploitation of capacity.

In case of freight and warehouse exchanges, we have to define a complex objective function. On a part of the total transport route, the freight tasks are transmitted together and then with the help of a combi terminal the freight tasks are transferred (multimodal transportation with rail/river, BA_ACO algorithm, Fig. 3).

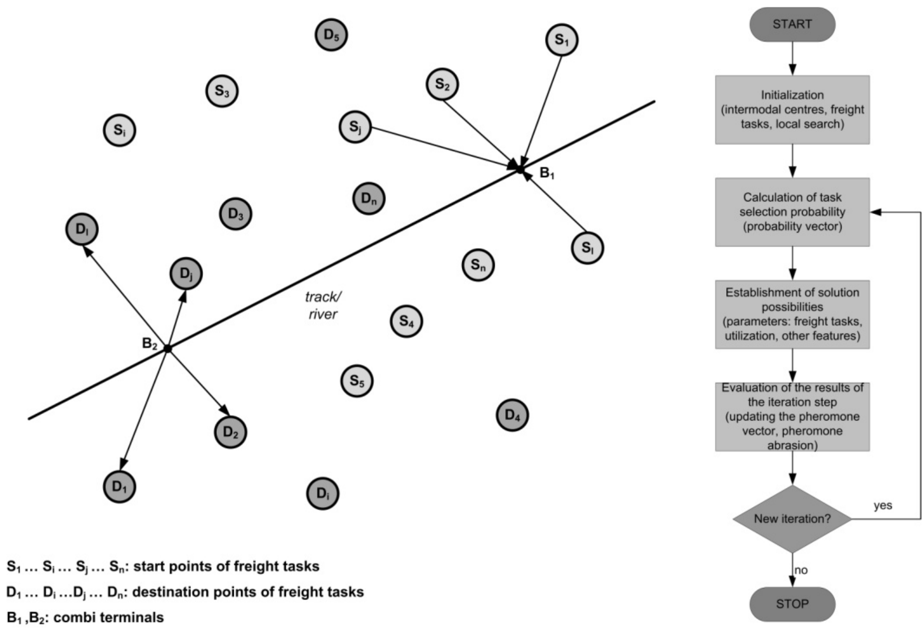


Fig. 3. Multimodal transport supported by freight and warehouse exchange: the model layout and the ant colony algorithm (BA_ACO)

The objective functions (maximum benefit, H, see equation (2)):

1. maximal use (K^{CF}) of the rail/river vehicle,
2. maximal total mileage reduction (F^{CF}, kilometre),
3. minimal transportation performance increase (Q^{CF}, ton*kilometre),
4. optimal demand of the surplus logistic services (RI^{BF}).

$$\mathbf{H=RI^{BF} \cdot K^{CF} \cdot F^{CF}/Q^{CF}=>MAX!} \quad (2)$$

This problem can be solved by ACO (ant colony optimization), which is an optimizing algorithm developed by Marco Dorigo [26] based on the modeling of the ants' social behavior. In nature ants search for food by chance, then if they find some, on their way back to the ant-hill they mark the way with pheromone. Other ants – due to the pheromone sign – choose the marked way with higher probability instead of accidental wandering. Shorter ways may be completed quicker, thus on these ways more pheromone will be present than on longer ones. After a while the amount of pheromone drops (evaporation), by this preventing sticking to local optimum.

In the electronic freight and warehouse exchange similar problem emerges as the ants' search for food: the target is the agile performance of freight/storage tasks offering the higher profit. There are some researches in this topic [27], [28], [29].

The ant colony algorithm usable in electronic freight and warehouse exchanges (BA_ACO) operates upon the following large-scale procedure (Fig. 3, [29]):

- Definition of input data:
  - starting point of optimum search (e.g. combi terminals, etc.),
  - narrowing down search space (local search): e.g. the selection of performable freight tasks depending on the distance compared to the combi terminals,
  - collection of the main features of the combined/non combined transport (mileage, transportation performance),
  - establishment of pheromone vector (the strength of the selection of freight tasks),
  - settling of profit vector (how much profit the selection of freight tasks will bring from the aspect of the route/solution).
- Calculation of task selection probability:
  - the probability that a freight task will be fulfil through combined transport (based on e.g. pheromone, quantity, distance, etc.),
  - a vector may be formed from the above-mentioned probabilities (probability vector).
- Establishment of solution possibilities:
  - establishment of random numbers, then selection of freight tasks upon probability vector, until the realization of the limiting conditions (e.g. capacity of train),
  - definition of the main features of the route (objective function parameters, equation (2)),
  - execution of the above-mentioned tasks in accordance with the number of ant colonies (e.g. ten ants = ten versions).
- Evaluation of the results of the iteration step:
  - filling in the profit vector: freight tasks by freight tasks, choosing the highest profit in aspect of the total route/solution and set it in to the current freight task, then updating the maximum profit reached in the iteration steps, if improvement was realized,
  - updating the pheromone vector (based on e.g. reached profit).
- Making new and new iteration steps (as long as further improvement cannot be reached, or after certain number of steps).

The algorithm (BA_ACO) was coded and tested in MS Visual Basic language, as well as formulas and its characteristics. In the course of checkout there have been executed lot of runs (see Fig. 4: box plot chart helps to evaluate the parameters of program /BA_ACO/ and the results). Based on this, we can change for example the number of solutions (ants), iterations or runs.

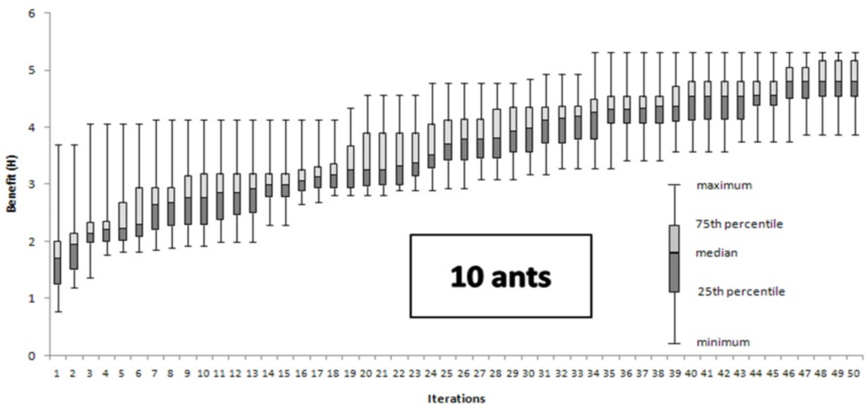


Fig. 4. Box plot (the changes of benefit; 10 ants /10 solution versions/, 40 runs, 50 iterations)

7 Conclusion

With the help of the presented methods, by the filtering of local optimums, an agile solution can be found shortly, which to freight/storage capacities/tasks selects freight/storage capacities/tasks. Through the coordination we are able to establish e.g. collecting-distributing routes, to organize back haul, and through this to reduce the number of vehicles. In this way, support of complex logistics processes will be possible (further research opportunities are based on e.g. [30]). In other words, freight and warehouse exchanges are one of the "simplest", but still the most efficient way of optimizing complex logistics processes and developing agile supply chains.

References

1. Christopher, M.: Logistics and Supply Chain Management. Financial Times/Pitman Publishing, London (1998)
2. Grzybowska, K.: Consistent supply chain - a way to increase the efficiency of metastructures and survival in a crisis. Change in Condition for Success (128), 319–326 (2010) (polish)
3. Grzybowska, K.: Constitutive elements of the supply chain. In: Pawłowski, E. (ed.) Operations and Logistics Management, pp. 77–91. Publishing House of Poznan University of Technology (2010)
4. Mentzer, J.T., DeWitt, W., Keebler, J.S., Min, S., Nix, N.W., Smith, C.D., Zacharia, Z.G.: Defining Supply Chain Management. Journal of Business Logistics 22(2), 1–25 (2001)
5. Rakovska, M.: Characteristics of supply chain management in Bulgaria. Research in Logistics & Production 1(2), 93–104 (2011)
6. Christopher, M.: The Agile Supply Chain: Competing in Volatile Markets. Industrial Marketing Management 29(1), 37–44 (2000)
7. Lee, H.L.: E-business and supply chain integration. Stanford Global Supply Chain, Forum (2001), `http://www.stanford.edu/group/scforum/Welcome/EB_SCI.pdf`

8. Wooldridge, M., Jennings, N.R.: Intelligent agents: Theory and practice. The Knowledge Engineering Review (10), 115–152 (1995)
9. Moyaux, T., Chaib-draa, B., D'Amours, S.: Supply Chain Management and Multiagent Systems: An Overview. In: Chaib-draa, B., Muller, J. (eds.) Multiagent Based Supply Chain Management. SCI, vol. 28, Springer, Heidelberg (2006)
10. Ambe, I.M.: Agile Supply Chain: Strategy for Competitive Advantage. Journal of Global Strategic Management (7), 5–17 (2010)
11. Reengineering the Foundation for B2B Information Exchange in the Supply Chain. A GT Nexus White Paper, pp. 1–7 (2011)
12. Venkatraman, N., Henderson, J.C.: Real Strategies for Virtual Organisations. Sloan Management Review (Fall), 33–48 (1998)
13. Yusuf, Y.Y., Gunasekaran, A., Adeleye, E.O., Sivayoganathan, K.: Agile supply chain capabilities: Determinants of competitive objectives. European Journal of Operational Research (159), 379–392 (2004)
14. Stevens, G.C.: Integrating the Supply Chains. International Journal of Physical Distribution and Materials Management 8(8), 3–8 (1989)
15. Grzybowska, K.: The development of the supply chain – not at the same pace at all times. Core and periphery enterprises in industrial networks. Economics Alternatives (1), 82–89 (2011)
16. Beynon-Davies, P.: E-business. Palgrave Macmillan, Basingstoke (2004)
17. Rayport, J.F., Jaworski, B.J.: Introduction to e-Commerce. McGraw-Hill, New York (2002)
18. Kovács, G.: The structure, modules, services and operational process of modern electronic freight and warehouse exchanges. Periodica Polytechnica Transportation Engineering 37(1-2), 33–38 (2009)
19. Winston, W.L.: Operations research I-II. Duxbury Press, Belmont (1994)
20. Ihde, T.: Dynamic alliance auctions - A mechanism for Internet-based transportation markets. Physica Verlag, Heidelberg (2004)
21. Song, J., Regan, A.C.: Approximation algorithms for the bid valuation and structuring problem in combinatorial auctions for the procurement of freight transportation contracts. Transportation Research Part B: Methodological 39(10), 914–933 (2005)
22. Kovács, G., Bóna, K., Duma, L.: Methodology and managerial lessons of tendering logistics hardwares. Acta Technica Jaurinesis Series Logistica 1(2), 237–246 (2008)
23. Saaty, T.L.: How to make a decision: the analytic hierarchy process. Interfaces 24(6), 19–43 (1994)
24. Saaty, T.L.: The analytic hierarchy process. University of Pittsburgh, Pittsburgh (1980)
25. Kovács, G., Grzybowska, K.: Logistics processes supported by freight and warehouse exchanges. In: Grzybowska, K., Golińska, P. (eds.) Selected Logistics Problems and Solutions, pp. 80–95. Publishing House of Poznan University of Technology, Poznan (2011)
26. Dorigo, M., Stützle, T.: Ant colony optimization. MIT Press, Cambridge (2004)
27. Bin, Y., Zhong-Zhen, Y., Baozhen, Y.: An improved ant colony optimization for vehicle routing problem. European Journal of Operational Research 196(1), 171–176 (2009)
28. Bell, J.E., McMullen, P.R.: Ant colony optimization techniques for the vehicle routing problem. Advanced Engineering Informatics 18(1), 41–48 (2004)
29. Kovács, G.: The ant colony algorithm supported optimum search in the electronic freight and warehouse exchanges. Periodica Polytechnica Transportation Engineering 39(1), 17–21 (2011)
30. Lénárt, B.: Automatic identification of ARIMA models with neural network. Periodica Polytechnica Transportation Engineering 39(1), 39–42 (2011)

Standards for Transport Data Exchange in the Supply Chain – Pilot Studies

Boguslaw Sliwczynski[1], Marcin Hajdul[1], and Paulina Golinska[2]

[1] Institute of Logistics and Warehousing, Estkowskiego 6, 61-755 Poznań, Poland
[2] Poznan University of Technology, Strzelecka 11, 60-965 Poznan, Poland
{boguslaw.sliwczynski,marcin.hajdul}@ilim.poznan.pl,
paulina.golinska@put.poznan.pl

Abstract. Cooperation in the supply chain requires a lot of information exchange between different companies. In order to achieve appropriate data exchange open communication standards are needed which are commonly agreed and approved. In the field of logistics a number of standards already defined. The aim of the paper is to compare existing communication standards to the market needs and expectations. Authors present conclusions from pilot survey conducted within selected group of transport users and logistics service providers regarding information exchange by planning and execution of transport services.

Keywords: communication standards, data exchange, transport services.

1 Introduction

In order to ensure continuity and efficiency of product flow to the client as well as promptness and reliability of deliveries, it is necessary to coordinate cooperation with partners who have appropriate delivery potential, who are able to ensure required reliability and flexibility and who are able to react fast. Integration of numerous management areas results from data exchange within the process of common planning and scheduling of partners' operating activities in the supply chain.

The intensive development of Business Intelligence and Competitive Intelligence tools, access to information from multi-dimensional data analysis [1] aggregated from various enterprise IT systems sources (usually in case of heterogeneous environments) has been significantly facilitated.

Numerous possible cooperation modes within supply chain provoke interest in easy and quick exchange of electronic data among potential transport providers and transport users [5]. Common access to electronic data relating to supply chains as well as transport services, electronic data exchange and electronic integration of planning, supply chains organisation and management of subcontractors' activities form the basis for the subsequent stage in the development of logistic outsourcing.

The aim of the paper is to compare developing by numerous EU-funded projects and industry led initiatives such as GS1, Common Framework, a communication standard to the market needs and expectations.

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 586–594, 2012.

A 'Common Framework' is a framework for interoperability [4]. It consists of roles, business processes, ontology and messages to support interoperability in the main logistics processes, providing the information infrastructure to support mode-independent transport planning, compliance, cargo consolidation and visibility. This means simplification, mode integration, more collaboration among SC business partners and authorities, new skills and, in the end, more competitiveness in logistics.

In this paper are presented conclusions from pilot survey, aiming at practical verification of the Common Framework, conducted within selected group of transport users and logistics service providers.

2 Data Standards as the Basis of Cooperation between Partners

Standardisation of requirements regarding data exchange between players on the transport services market, including effective and efficient management by service providers, requires exchange of agreed information which is unambiguously defined and interpreted by all players on the services market. These standards should be: commonly and always available free of charge and must remain accessible to in the future. Moreover standards should be documented in all exploitation areas.

There are over 400 standardisation initiatives recorded in the world. Standardisation in information systems covers many of their component parts used in electronic data exchange servicing:

- identification of the transaction object (product, service) and subject – e.g. GS1 standard: product identification – GTIN (Global Trade Item Number), location identification - GLN (Global Location Number), asset identification – GIAI (Global Individual Asset Identifier), GRAI (Global Returnable Asset Identifier),
- unambiguous identification of information elements – e.g. country - ISO Country Codes, date, time - UN Numerical Representation of Dates, Times and Periods of Time, currency - UN Alphabetic Code of the Representation of Currencies, UN Codes for Units of Measurement used in International Trade (e.g. 1200 units), UN LOCODE - Code for Ports and other Locations, ISO/UN Trade Data Element Directory,
- document layouts and structures (layout keys) – e.g. UN Layout Key; including UN Aligned Invoice Layout Key, EDIFACT; EDI-XML GS1
- arrangement of codes in documents - basis: UN Location Codes,
- unique identification code methodology – basis: UN Unique Identification Code Methodology,
- e-commerce self-regulatory instruments – basis: UN E-Commerce Self-Regulatory Instruments (Codes of Conduct),
- security and certification of information - W3C XML Signature, W3C XML Encryption, XACML (Extensible Access Control Markup Language), SAML (Security Assertions Markup Language), XKMS (XML Key Management), Web Services Flow Language, XML Digital Signature,

- description of the electronic services process in the area of public services – basis UN/CEFACT and OASiS ebXML Business Process Specifcation Schema and ebXML Catalog of Common Business Processes: BPEL4WS Business Process Execution Language for Web Services;
- description of the electronic communication system profile in the area of public services – basis: ebXML Collaboration Protocol Profile and Agreement Specification: UN/CEFACT and OASiS,

Data exchange standardisation services in the area of e-commerce also include:

- use of open internet communication protocols syntaxes (SOAP, WSDL, UDDI, ebXML Messaging, RNIF, etc.),
- use of standards for security and certification of exchanged messages – e.g. W3C XML Signature, W3C XML Encryption, XACML (Extensible Access Control Markup Language), SAML (Security Assertions Markup Language), XKMS (XML Key Management), Web Services Flow Language, XML Digital Signature, Business Transaction Protocol and SOAP extentions),
- use of technical and technological standard translation adapters of data exchange in descriptions of electronic data exchange systems' profiles,
- use of reliable messaging mechanisms independent of the transport protocol - which ensure reliable (e.g. once and only once) delivery of data and documents. Basic elements of such mechanism include:
- persistent storage - database of received documents,
- mechanisms to send confirmations and detect duplicates,
- an automated mechanism to re-send documents after timeout or in case of lacking confirmation,
- specifications of communications which check statuses of received documents and data packages,
- moreover, standardisation (currently, mostly in accordance with bilateral agreements) includes:
- reliability of synchronic and asynchronic exchange,
- definitions of own reporting and error correction mechanisms,
- definitions of own mechanism for identification and authorisation of parties exchanging data and electronic documents.

In addition to this a number of EU funded research and business development projects have been addressing the issues of information and communication technologies in transport and logistics. Traditionally, these projects have been quite autonomous and there has been little coordinated contact between the projects. This has, however, now changed. The partners in a number of projects have realised that there are project benefits that may be exploited from better cooperation. This view is also shared by the EU Commission. As industrial developments and the research community have reached similar conclusions, a joint initiative has been taken to improve interoperability considerably by developing Common Framework for exchange of information between ICT systems in transport and logistics. The Common Framework has a vision to become an umbrella of all major existing communication standards.

The Common Framework supports interoperability between commercial actors and communication to authorities – to make the best possible use of the available transportation infrastructure, provide appropriate supply chain security, and support compliance requirements. Authors concentrated only on the part of the Common Framework which is dedicated to relation between transport user and transport service providers, mainly in European trade-lines. The main issue that appear by coordination of process in the virtual cooperation network is independency of technology in order to support the possibility of temporary relations and reconfiguration. In case of the transport process the requirements for the Common Framework have been identified [2], as:

- support multimodality (co-modality),
- be stable and easy to refine and expand,
- be future-oriented (independent of current solutions),
- provide a total picture (supporting transparency, management, and security),
- facilitate hiding of complexity (abstraction, simplification),
- focus on interoperability (not on inner parts of systems),
- independent of technology,
- facilitating interaction with existing standards (to help protect investments already made in legacy and other systems).

Figure 1 presents basic information groups necessary to plan and supervise as well as monitor and report the provision of transport services and quick reaction possibilities in case deviations from the plan occur.

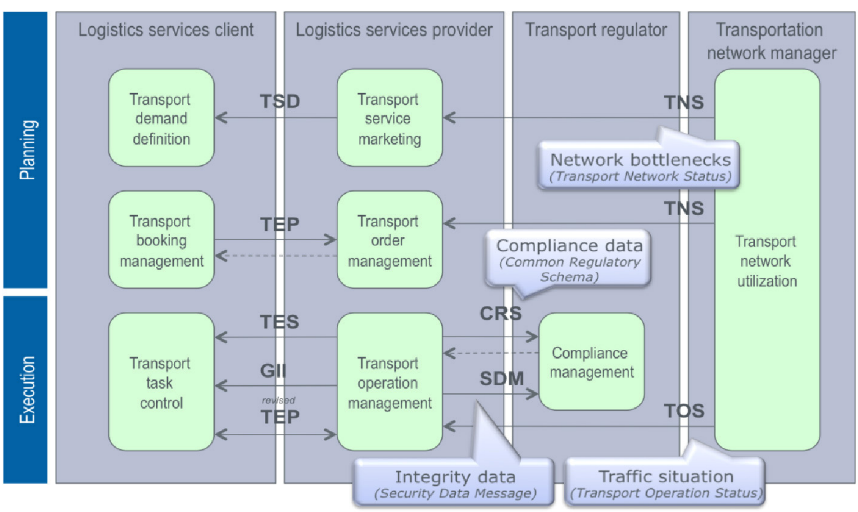


Fig. 1. Standard groups of information exchange between supply chain players in the process of planning and provision of transport services, Source: [2]

Table 1 presents the purpose and scope covered by particular information groups which are used in planning and execution of transport services in the Common Framework.

Table 1. Examples of standard messages in transport process developed within Common Framework

Information Description	**Purpose**	**Scope**
TSD (Transport Service Description)	TSD is used for a description of transport services suitable for automatic detection. It applicable for transfer of, operations in terminals, discharging, and additional services.	Description of service including: - scope of services, - prices, - type of cargo (food, electronics, ADR ect.), - type of packing units (pallets, ect)
TEP (Transport Execution Plan)	TEP describes all the information needed related to the execution of a transport service between transport user and transport provider.	Remarks about the execution of the transport service, special conditions, delivery time, delivery location.
TES (Transport Execution Status)	TES provides information about the progress of the transport and of the cargo condition.	Description of the freight status (e.g. current freight conditions).
CRS (Common Regulatory Schema)	CRS provides information regarding regulation and monitoring of proper execution of transport services to the supervising authorities	Iindication of the container specific positioning information, using the Security Data Message
SDM (Security Data Message)	SDM gives information required by law on account of security of the transport system, traffic, transport services,	Definition a standard format for data provided Container Security Device (sealed load unit) with a reference to movement reference nr assigned by customs authorities, other information required by Import Control, System.
GII (Good Item Itinerary)	Provides information on the movement of goods on the whole way including transhipments	Description of the complete itinerary for a given goods item, including planned, estimated, and actual times for departure and arrival for each service.
TOS (Transport Operation Status)	TOS assists in establishing the best possible arrival time estimates.	Reports on transport execution status including deviations from planned routes and timetables.
TNS (Transportation Network Status)	TNS points out traffic, including information from the different transport modes.	Information on vehicle movements, obstacles, traffic jams.

3 Results of Research on the Use of Transport Information Standards in Supply Chain Enterprises

The pilot survey was conducted among group of about 30 companies. The companies represent both transport users and transport providers. The research aims to compare the required data standards with the information which is actually used in the process of planning and provision of logistics and transport services. The research highlights the information exchange between the client/transport service recipient and transport service provider (fig. 2).

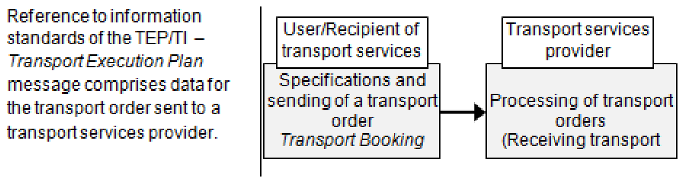


Fig. 2. The Common Framework messages selected for the study

The group of respondents was asked about the application of common standards. The results are presented in fig.3.

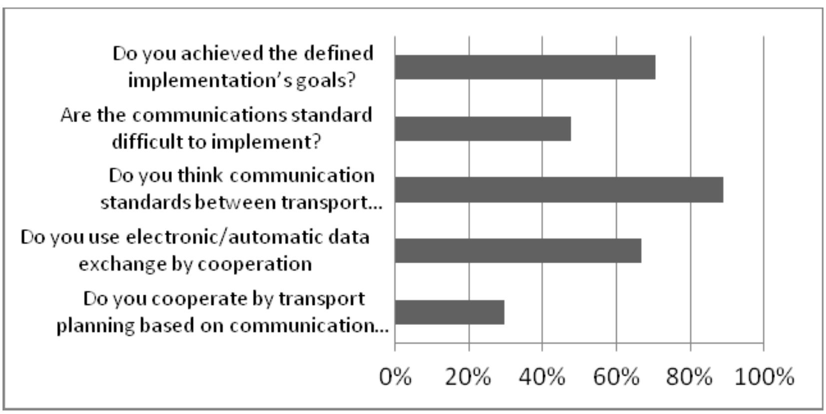


Fig. 3. Positive answers regarding the application of common communication standards

At present the analysed companies (transport providers and users) don't use very often common communications standards (30% of positive answers). Most of the analysed transport providers use electronic data exchange (c.a.65%). In the open question the respondents have identified the main barriers which appear by implementation of communications standards:

- high cost of implementation,
- long time needed for implementation,
- human factor (difficulties with the staff training),
- technical difficulties (system errors) in the initial stage of implementation.

Most of companies (almost 90%) have declared that common communication standards are needed.

Table 2 presents selected results of the comparison of information actually exchanged in the Polish transport services market in relation to international information standards regarding planning and provision of such services.

Table 2. Comparison of information sent by transport users in practice and required by standard

The Common Framework standards for planning of transport services		Real information used in practice by companies from ECR Poland on the market of transport services	
Data group	Detailed specifications	Pilot research results	Sample survey content
Consignment	Consignment ID	Non-standard	order no.; consignment note no.;
	Delivery Requirements	Standard for limited load units	stack pallets; operator renders the services of storage, picking and transport
	Number Of Transport Items	Non-standard	number and weight of parcels; number and weight of pallets; number and weight of containers; Load - GTIN code + our goods number + number of the goods delivered, number of loading meters of goods
	Sender (Company Code, Name, Details)	Yes	City (typical error – the same city have different names in different databases, zip code (sometimes the same location have different zip codes in different databases)
	Receiver (Company Code, Name, Details)	Yes	City (typical error – the same city have different names in different databases, zip code (sometimes the same location have different zip codes in different databases)
Contract Reference	Contract (ID, Name, Description; Include agreement of subcontractor services - transport, storage handling, etc.)	Non-standard	Order no. (it can be at the SKU level, description of goods, number) Contract number, INCOTERMS 2010, delivery terms: FCA Payer for transport
	Declaration Date, Contract Expire Date, Min and Max Quantity	No data	
	Nomination Time Deadline for booking (in days prior transport start)	Yes	Containers booking directly at the shipowner – e.g. for week 25

Table 2. (*continued*)

Document Preferences	Document Type eg. IMDG – Sea, ADR – Road, RID – Rail, IATA – Air; BL, dangerous goods description, customs declaration, etc.	Yes	Type/ID of transport document, loading documents, ownership, insurance Details to be included in documents: Number of consignment note Specially adapted EDI messages with information on the place and time of delivery. It is necessary to place the number of bulk packages (if there are any) and number of pallets in transport documents. Logistic information is agreed upon in advance with the operator;

4 Conclusions

The analysis of research results regarding the comparison of the scope of information sent by transport users has led to a conclusion that compliance with the Common Framework for ICT in transport & logistics dedicated to the process of planning and provision of transport services reaches approx. 40%. The data standard model has been approved for the service commissioner - service provider relationship, however, authors have observed that there are currently numerous types of services between users of the transport systems, which influences the scope of sent information, like:

- outsourcing transport services to an external subcontractor – the supplier manages the fleet,
- transport commission – the loader manages and designs requirements for provided vehicles and ensures the return run (complex route data),
- outsourcing distribution to an external subcontractor,
- service provider executes the transport using a container ordered by the loader (interrelated service with two reception/loading points - container and freight - and two delivery points); the service provider delivers the container as an integral part of services covered by the contract/ order.

The research has allowed the authors to recognise a few areas of significant differences between the practical application and assumptions of the Common Framework dedicated to the process of planning and provision of transport services. In practical terms, the GS1 [3] standard is applied in a much wider scope of exchanging information relating to provision of the transport service. It includes: EANCOM communication standards, standards for location identification based on the GLN code, identical for all forms of transport, identification standards regarding products, load and transport items.

By contrast, the TES communication (in the location description status) uses among others: UNLoc code, postal area zip code. The TSD communication is in practical applications sent in two directions:

- the client sends transport requirements: Transport Service Demand,
- the service provider sends a transport service offer: Transport Service Description.

Analysed version of the Common Framework, TEP/TI transport commission did not include details of given products, data is limited to freight description. However, in the Polish transport services market practice detailed data refer to products and their identifiers (codes). They provide a link between data in the order and in the transport commission. In compliance with information standards, detailed product data is sent within GII/TIS data (Good Item Itinerary / Transport Item Status) - information on the movement of goods, sent by the provider of transport services (possibility to connect data from numerous services in road, rail, sea or air transport) whereas in the transport services market practice product data is already sent at the TEP/TI transport commission stage. In order to reach the goals of European Transport Policy is needed to increase the efficiency of transport services and infrastructure's utilization with information systems and market-based incentives [6]. Some of the data used in practice may be allocated to certain information standard areas while it is difficult to allocate the majority of sent information to adequate fields in accordance with information standards. For example TSD data does not include specifications of serviced documents, the serviced transport relation, etc. Considering the above, it is advisable to verify and adopt in commercial practice the standard described in 'Common Framework for Information and Communication Systems In Transport and Logistics' which is strongly supporting by the European Commission and still developing by different research project like Comcis, eFreight or i-Cargo.

References

[1] Sołtysik-Piorunkiewicz, A.: Controlling in Organisation and Management. Computerisation Concept. Humanitas Publishing House, Sosnowiec (2009)

[2] Pedersen, T.J., Paganelli, P., Knoors, F.: One Common Framework for Information and Communication Systems in Transport and Logistics, DiSCwise project deliverable, Brussels (2010)

[3] GS1. GS1 standards in transport and logistics. GS1 Global Office, Brussels (2010)

[4] `http://www.discwise.eu` (accessed December 18, 2011)

[5] Hajdul, M.: Model of coordination of transport processes according to the concept of sustainable development. LogForum 3(21), 45–55 (2010)

[6] Golinska, P., Hajdul, M.: European Union Policy for sustainable transport system – challenges and limitations. In: Golinska, P., Hajdul, M. (eds.) Sustainable Transport, pp. 3–20. Springer, Heidelberg (2012)

Smooth Component Analysis and MSE Decomposition for Ensemble Methods

Ryszard Szupiluk[1,2], Piotr Wojewnik[1,2], and Tomasz Ząbkowski[1,3]

[1] Polska Telefonia Cyfrowa Ltd., Al. Jerozolimskie 181, 02-222 Warsaw, Poland
[2] Warsaw School of Economics, Al. Niepodleglosci 162, 02-554 Warsaw, Poland
[3] Warsaw University of Life Sciences, Nowoursynowska 159, 02-787 Warsaw, Poland
{rszupiluk,pwojewnik,tzabkowski}@t-mobile.pl

Abstract. The paper is addressed to economic problems for which many different models can be proposed. In such situation the ensemble approach is natural way to improve the final prediction results. In particular, we present the method for the prediction improvement with ensemble method based on the multivariate decompositions. As a method for model results decomposition we present the smooth component analysis. The resulting components are classified as destructive and removed, or as constructive and recomposed. The classification of the components is based on the theoretical analysis of *MSE* error measure. The robustness of the method is validated through practical experiment of energy load consumption in Poland.

Keywords: Ensemble Methods, Multivariate Decompositions, Smooth Component Analysis.

1 Introduction

The term *model* plays an important role in economic sciences. There are different meanings and scopes of the *model* which depend on the specific application areas, economic schools or even the authors. The common feature linking these different understandings of the model is the awareness that the model presents only a given perspective of the real phenomenon and it is considered only as a useful analytical representation of the given problem. In this sense we can find that any real process or phenomenon may be represented by different models. The current economic crisis clearly demonstrates the danger of the situation in which we are too much focused on one dominant method or paradigm of phenomenon interpretation. For instance, current description of the financial markets is dominated by models which are based on human behavior, while in fact up to 70% percent of transactions are generated by automated systems [16].

Based on the economic theory, the situation is very complex because there are many theories and models which are related to the same phenomena, but often operating in separate and relatively hermetic economic schools. It seems that one of the fundamental challenges facing modern economic theory is to develop an effective

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 595–603, 2012.

method to combine information coming from different theoretical approaches. At the same time it looks that combination of theory is not possible due to the original assumptions underlying at various economic approaches. For example, it is difficult to bring together the extreme a priori and deductive Austrian School with inductive modeling approach represented by agents or data mining. It seems that the main area in which this synthesis is possible are the final results derived from models and theories. In particular, this refers to the quantitative results. The advantages of models aggregation are particularly evident in models which aim at the instrumental effects of their actions. This applies especially to agent models, machine learning, data mining and econometric models, what in fact involves all approaches that are based on mathematical methods of data exploration [5,11]. Let's bear in mind that although the method of models aggregation are currently popular research area, the aggregation of different models with a completely different methodological approaches is not obvious. Most of the existing aggregation methods such as boosting, bagging, or stacked regression require quite restrictive assumptions about the parameters and the structure of aggregated models and distributions of the variables as well [1,6,15]. In most cases, aggregation concerns models of the same structure and even the same models but estimated on other subsets of the training data.

In this paper we develop the approach of aggregation based on blind signal separation methods. In this concept, we focus rather on the physical meaning and interpretation of the data (variables) than on their formal mathematical properties. We assume that the results generated by different prediction models are certain physical components that disrupt the prediction. They may be related to the inaccuracy, inadequacy or noisy input data, but they can also be the result of improper models specification or wrong learning algorithm choice. As a result, we can assume that the models results are the combination of certain constructive and, from the other side, destructive components which are responsible for the prediction errors. Therefore, separation and then elimination of these destructive components should bring prediction improvement [13].

In this paper we apply smooth component analysis method to find the latent components [14] and the next step is to classify them as destructive or constructive. For this task we propose analysis of second order statistics – mean squared error criterion, *MSE*. The validity of the conception is confirmed by practical experiment with energy load prediction from Polish market [8].

2 Prediction Results Model

In this paper we assume that after learning various models we have a set of prediction results. For simplicity, in further consideration we assume that our models results x_i and target p are centered. In practice, it means that before models integration we remove mean values from prediction results and target, and after integration we add this values.

We collect particular prediction results x_i, $i=1,...,m$, in one multivariate variable $\mathbf{x}(k)=[x_1(k),...,x_m(k)]^T$. Now we assume that prediction results $x_i(k)$ is a mixture of the latent components: constructive $\hat{s}_j(k)$ associated with the predicted variable, and destructive $s_j(k)$ associated with the inaccurate and missing data, imprecise estimation, distribution assumptions etc. We assume the relation between observed prediction results and latent components to be represented as

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k)\,, \tag{1}$$

where $\mathbf{s}(k) = [\hat{s}_1(k),...,\hat{s}_i(k), s_{i+1}(k),...,s_m(k)]^T$, matrix $\mathbf{A} \in R^{m\times m}$ represents the mixing system. The relation (1) stands for decomposition of prediction results **x** into latent components matrix **s** and mixing matrix **A**. Our aim is to find **s** and **A**, reject its destructive part (replace some signals $s_j(k)$ with zero) and next mix the constructive components back to obtain improved prediction results:

$$\hat{\mathbf{x}}(k) = \mathbf{A}\hat{\mathbf{s}}(k) = \mathbf{A}[\hat{s}_1(k),...,\hat{s}_i(k), 0_{i+1},...,0_m]^T \,. \tag{2}$$

The crucial point of the above concept is proper **A** and **s** estimation. The most adequate methods to solve the problem seem to be the Blind Signal Separation (BSS) techniques [3,7,12]. The crucial point of the above concept is proper **A** and **s** estimation. This problem can be described as Blind Signal Separation task which aims to find such matrix $\mathbf{W} = \mathbf{A}^{-1}$ that

$$\mathbf{s}(k) = \mathbf{W}\mathbf{x}(k)\,. \tag{3}$$

The BSS methods explore different properties of data like: independence [2,3,7], decorrelation [3], sparsity [10], smoothness [3,14], non-negativity [9] etc. In this paper we focus on Smooth Component Analysis what is adequate for data with temporal structure.

Smooth Component Analysis (SmCA) is a method for the smooth components identification in a multivariate variable [3,14]. For *N*-observation signals with temporal structure we propose a following smoothness measure

$$P(s) = \frac{\dfrac{1}{N}\sum\limits_{k=2}^{N} | s(k) - s(k-1) |}{\max(s) - \min(s) + \delta(\max(s) - \min(s))}\,, \tag{4}$$

where symbol $\delta(.)$ means zero indicator function – valued at 0 everywhere except 0, where the value of $\delta(.)$ is 1. Measure (4) has straightforward interpretation: it is maximal when the changes in each step are equal to range (maximal possible change during one period), and is minimal when data are constant. The possible values vary from 0 to 1. Zero indicator $\delta(.)$ term is introduced to avoid dividing by zero.

The components are taken as linear combination of signals x_i and should be as smooth as possible. Our aim is to find such matrix $\mathbf{W}=[\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n]$ that for $\mathbf{s} = \mathbf{W}\mathbf{x}$ we obtain $\mathbf{s} = [s_1, s_2, ..., s_m]^T$ where s_1 maximize $P_1(s_1)$ so we can write

$$\mathbf{w}_1 = \arg\max_{\|\mathbf{w}\|=1}(P(\mathbf{w}^T\mathbf{x})) \quad . \tag{5}$$

Having estimated the first $k-1$ smooth components the next one is calculated as least smooth component of the residual obtained in Gram-Schmidt orthogonalization [4]:

$$\mathbf{w}_k = \arg\max_{\|\mathbf{w}\|=1}(P(\mathbf{w}^T(\mathbf{x} - \sum_{i=1}^{k-1}\mathbf{s}_i\mathbf{s}_i^T\mathbf{x}))) \quad , \tag{6}$$

where $\mathbf{s}_i = \mathbf{w}_i^T\mathbf{x}, i = 1 \ldots k$. As the numerical algorithm for finding $\mathbf{w}_n$ we can employ the conjugate gradient method with golden search as a line search routine. The algorithm outline for initial $\mathbf{w}_i(0) = rand$, $\mathbf{p}_i(0) = -\mathbf{g}_i(0)$ is as follows:

1. Identify the indexes l for extreme signal values:

$$\mathbf{w}_k = \arg\max_{\|\mathbf{w}\|=1}(P(\mathbf{w}^T(\mathbf{x} - \sum_{i=1}^{k-1}\mathbf{s}_i\mathbf{s}_i^T\mathbf{x}))) \quad , \tag{7}$$

$$l^{\max} = \arg\max_{l\in 1\ldots N} \mathbf{w}_i^T(k)\mathbf{x}(l) \quad , \tag{8}$$

$$l^{\min} = \arg\min_{l\in 1\ldots N} \mathbf{w}_i^T(k)\mathbf{x}(l) \quad . \tag{9}$$

2. Calculate gradient of $P(\mathbf{w}_i^T\mathbf{x})$:

$$\mathbf{g}_i = \frac{\partial\, P(\mathbf{w}_i^T\mathbf{x})}{\partial\, \mathbf{w}_i} = \frac{\sum_{l=2}^{N}\Delta\mathbf{x}(l)\cdot sign(\mathbf{w}_i^T\Delta\mathbf{x}(l)) - P(\mathbf{w}_i^T\mathbf{x})\cdot(\mathbf{x}(l^{\max}) - \mathbf{x}(l^{\min}))}{\max(\mathbf{w}_i^T\mathbf{x}) - \min(\mathbf{w}_i^T\mathbf{x}) + \delta(\max(\mathbf{w}_i^T\mathbf{x}) - \min(\mathbf{w}_i^T\mathbf{x}))} \quad , \tag{10}$$

where $\Delta\mathbf{x}(l) = \mathbf{x}(l) - \mathbf{x}(l-1)$.

3. Identify the search direction (Polak-Ribiere formula [4])

$$\mathbf{p}_i(k) = -\mathbf{g}_i(k) + \frac{\mathbf{g}_i^T(k)\left(\mathbf{g}_i(k) - \mathbf{g}_i(k-1)\right)}{\mathbf{g}_i^T(k-1)\mathbf{g}_i(k-1)}\mathbf{p}_i(k-1) \quad , \tag{11}$$

and calculate the new weights:

$$\mathbf{w}_i(k+1) = \mathbf{w}_i(k) + \alpha(k)\cdot\mathbf{p}_i(k) \quad , \tag{12}$$

where $\alpha(k)$ is found in golden search.

The above optimization algorithm should be applied as a multistart technique with random initialization.

The orthogonalization processes (6) after each component identification ensures the components to be ordered by their smoothness with correlation matrix given by:

$$\mathbf{R}_{ss} = E\{\mathbf{s}\mathbf{s}^T\} = \mathbf{D}, \tag{13}$$

where **D** is a diagonal matrix. The property (7) will be explored for destructive component identification where prediction is scored by *MSE* criterion.

3 Destructive Components Identification

The *MSE* is one of the most popular criterion for model scoring. In our case we can describe MSE_i for each x_i as

$$MSE_i = E\{(p - x_i)^2\}, \tag{14}$$

where p is target variable. According to our assumptions the x_i can be expressed as linear combination of the latent components, what leads us to

$$MSE_i = E\left\{\left(p - \sum_{j=1}^{m} a_{ij} s_j\right)^2\right\} = E\left\{p^2 + \sum_{j=1}^{m}\sum_{l=1}^{m} a_{ij} a_{il} s_j s_l - 2p\sum_{j=1}^{m} a_{ij} s_j\right\}. \tag{15}$$

After SmCA the latent components are decorrelated and we have $E\{s_i s_j\} = 0$, so:

$$MSE_i = \sigma_p^2 + \sum_{j=1}^{m}\left(a_{ij}{}^2\sigma_{s_j}^2 - 2a_{ij}\rho_{p,s_j}\right). \tag{16}$$

where $\sigma_p^2 = E\{p^2\}, \sigma_s^2 = E\{s^2\}$ and $\rho_{p,s_j} = E\{ps_j\}$.

Presented MSE_i calculation for x_i prediction explains what happens after elimination of $a_{ij}s_j$ from x_i for every j=1,…,m. Namely, if the condition

$$\sigma_{s_j}^2 \le \frac{2\rho_{p,s_j}}{a_{ij}}, \tag{17}$$

holds, we can expect reduction of MSE_i by value

$$\vartheta_{ij} = a_{ij}{}^2\sigma_{s_j}^2 - 2a_{ij}\rho_{p,s_j}. \tag{18}$$

Therefore the value ϑ_{ij} can be used for choosing the component s_j reducing MSE_i by highest number.

4 Practical Experiment

To verify the validity of the concept we analyze the real-world-problem of energy load prediction from Polish energy market. We build six neural networks with different learning methods (delta, quasi-Newton, Levenberg-Marquard) to forecast the hourly energy consumption in Poland in 24 hours basing on 86400 observations: energy demand from last 24 hours, month, day of the month, day of the week, and holiday indicator. The test1 data set (43200 observations) is used to estimate the decomposition matrices **W**, **A** and to calculate the expected values of MSE reduction ϑ_{ij} . In the final phase we use test2 data set (43200 observations excluded from the previous analysis) to calculate the MSE reduction obtained after physical elimination of particular components.

The quality of the neural models on the test1 data set is presented in Tab. 1.

Table 1. MSE of the neural models on the test1 data set

MLP, Learning	28:11:1, Quasi-Newton	28:11:1, Delta	28:12:1, Levenberg-Marquard	28:13:1, Levenberg-Marquard	28:12:1, Levenberg-Marquard	28:10:1, Levenberg-Marquard
MSE [$\times 10^5$]	5,76	5,96	6,46	6,40	6,52	6,55

The smooth components analysis applied to the prediction results from the test1 data gives **A**, **W** and the components presented in Fig. 1.

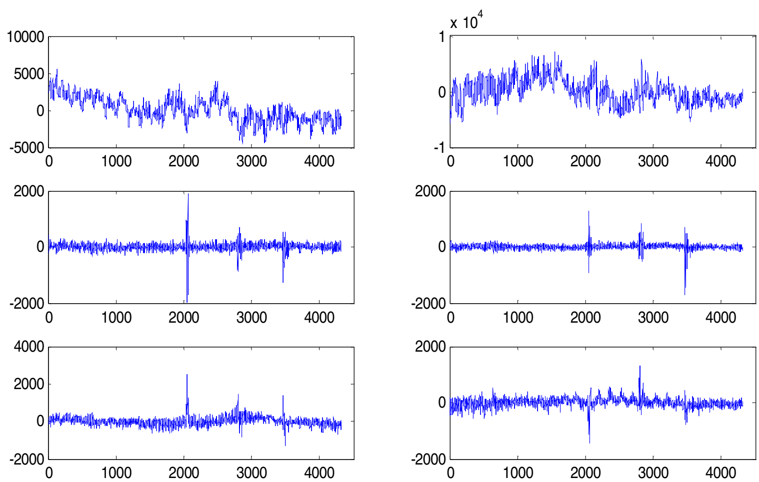


Fig. 1. Smooth components obtained from the test1 data

The smoothness $P(s_j)$ of identified signals is presented in Tab. 2

Table 2. Signal smoothness $P(s_j)$ for the test1 data

Component	s_1	s_2	s_3	s_4	s_5	s_6
$P(s_j)$	0,0287	0,0293	0,0363	0,0371	0,0405	0,0649

According to (12) for each model x_i, i=1,…,6, and for each signal s_j, j=1,…,6, we calculate expected level of MSE reduction ϑ_{ij}, see Tab. 3. In particular the component s_1 is constructive, because its elimination should increase the *MSE* of each model. The component s_5 is destructive; therefore, we expect that its elimination should decrease the MSE of each model.

Table 3. Expected MSE reduction ϑ_{ij} calculated on the test1 data set. Positive values denote value of MSE reduction possible to be obtained after elimination of s_j from x_i

ϑ_{ij} [×10_5]	28:11:1, Quasi-Newton	28:11:1, Delta	28:12:1, Levenberg-Marquard	28:13:1, Levenberg-Marquard	28:12:1, Levenberg-Marquard	28:10:1, Levenberg-Marquard
s_1	-58,19	-58,47	-58,36	-58,44	-58,31	-58,33
s_2	-9,72	-9,15	-8,07	-8,09	-8,08	-8,04
s_3	0,05	0,1	0,05	0,08	0	0
s_4	0,04	0,08	0,03	0	0	0,06
s_5	0,82	0,45	0	0,04	0	0
s_6	0,02	0,13	0,03	0	0,03	0,03

In Table 4 we present the quality of the models for the test2 data.

Table 4. Quality of the primary models on the test2 data set

MLP, Learning	28:11:1, Quasi-Newton	28:11:1, Delta	28:12:1, Levenberg-Marquard	28:13:1, Levenberg-Marquard	28:12:1, Levenberg-Marquard	28:10:1, Levenberg-Marquard
MSE [×10^5]	7,61	8,08	5,75	5,75	5,22	5,33

We decompose the test2 data using **W** matrix estimated from test1, physically eliminate the components and mix the signals back using **A**. In Table 5 we can observe obtained *MSE* reduction (ΔMSE_{ij}).

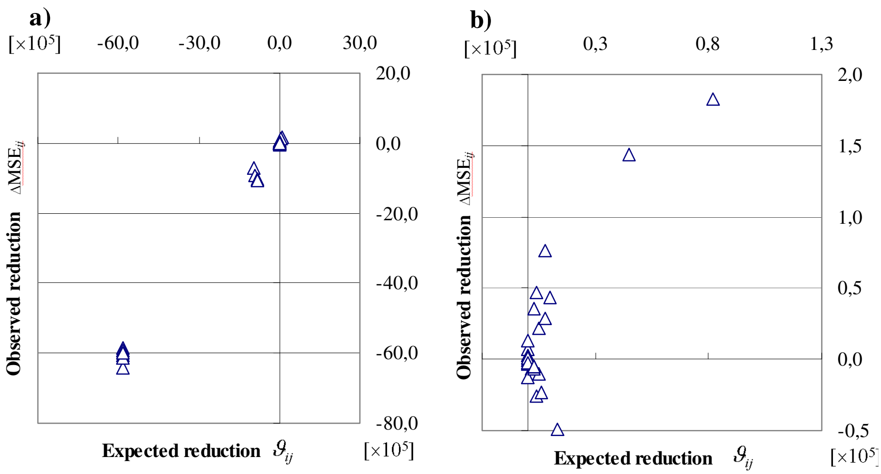


Fig. 2. Expected reduction ϑ_{ij} versus obtained values ΔMSE_{ij}: a) for every (x_i,s_i) combination, b) for the case of positive ϑ_{ij} values

Table 5. Positive values denote *MSE* reduction obtained after physical elimination of component s_j from prediction x_i in the test2 data set

ΔMSE_{ij} [$\times 10^5$]	28:11:1, Quasi-Newton	28:11:1, Delta	28:12:1, Levenberg-Marquard	28:13:1, Levenberg-Marquard	28:12:1, Levenberg-Marquard	28:10:1, Levenberg-Marquard
s_1	-58,66	-64,25	-60,49	-61,2	-59,2	-59,78
s_2	-6,97	-9,13	-10,24	-10,11	-10,67	-10,62
s_3	0,22	0,43	-0,1	0,29	0,03	-0,03
s_4	0,47	0,76	0,36	0,07	0,13	-0,23
s_5	1,83	1,44	-0,13	-0,26	0,01	-0,02
s_6	-0,07	-0,49	-0,07	-0,01	-0,05	-0,05

In Fig. 2 we present the observed ΔMSE_{ij} in comparison to the expected ϑ_{ij}.

The practical experiment proved validity of the concept of *MSE* reduction. In particular, we observed correlation between ϑ_{ij} and ΔMSE_{ij} at the level of 0.99.

5 Conclusions

In this paper we consider the integration of the information generated by different models using smooth components analysis. For the *MSE* criterion we present the theoretical background for efficient classification of the latent components. The results from the experiment confirm the rationality of the approach.

We mainly focus on the model's results decomposition based on smooth component analysis, but the above identification method can be addressed to wide area of data exploration models like agent and multi-agent models, simulations or machine learning systems. In particular, presented approach can be applied for trading systems (future research). We see the need of the models that can identify the fundamental factors influencing the stock market environment. Unfortunately, these factors are often hidden or mixed with noises. Therefore, a fundamental problem in financial market modeling is to estimate the main trends and to separate the general market dependencies from the individual behavior of a given financial instrument. This leads directly to the issue of data decomposition and interpretation of the underlying hidden components what in fact corresponds to our research presented in this paper.

References

1. Breiman, L.: Bagging predictors. Machine Learning 24, 123–140 (1996)
2. Cardoso, J.F.: High-order contrasts for independent component analysis. Neural Computation 11, 157–192 (1999)
3. Cichocki, A., Amari, S.: Adaptive Blind Signal and Image Processing. John Wiley, Chichester (2002)
4. Golub, G.H., Van-Loan, C.F.: Matrix Computations. Johns Hopkins (1996)
5. Haykin, S.: Neural networks: a comprehensive foundation. Macmillan, New York (1994)
6. Hoeting, J., Modigan, D., Raftery, A., Volinsky, C.: Bayesian model averaging: a tutorial. Statistical Science 14, 382–417 (1999)
7. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley (2001)
8. Lendasse, A., Cottrell, M., Wertz, V., Verdleysen, M.: Prediction of Electric Load using Kohonen Maps – Application to the Polish Electricity Consumption. In: Proc. of Am. Control Conf., Anchorage AK, pp. 3684–3689 (2002)
9. Lee, D.D., Seung, H.S.: Learning of the parts of objects by non-negative matrix factorization. Nature 401, 788–791 (1999)
10. Li, Y., Cichocki, A., Amari, S.: Sparse component analysis for blind source separation with less sensors than sources. In: Fourth Int. Symp. on ICA and Blind Signal Separation, Nara, Japan, pp. 89–94 (2003)
11. Mitchell, T.: Machine Learning. McGraw-Hill, New York (1997)
12. Stone, J.V.: Blind Source Separation Using Temporal Predictability. Neural Computation 13(7), 1559–1574 (2001)
13. Szupiluk, R., Wojewnik, P., Zabkowski, T.: Model Improvement by the Statistical Decomposition. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 1199–1204. Springer, Heidelberg (2004)
14. Szupiluk, R., Wojewnik, P., Ząbkowski, T.: Smooth Component Analysis as Ensemble Method for Prediction Improvement. In: Davies, M.E., James, C.J., Abdallah, S.A., Plumbley, M.D. (eds.) ICA 2007. LNCS, vol. 4666, pp. 277–284. Springer, Heidelberg (2007)
15. Yang, Y.: Adaptive regression by mixing. Journal of American Statistical Association 96, 574–588 (2001)
16. ForexNewsNow: Automated Trading: Pros and Cons (published July 7, 2011), `http://forexnewsnow.com/forex-brokers/automated-trading-pros-and-cons/`

Discussion of the Competence Management Models for Education Context

Przemysław Różewski

West Pomeranian University of Technology in Szczecin,
Faculty of Computer Science and Information Systems,
ul. Żołnierska 49, 71-210 Szczecin, Poland
prozewski@wi.zut.edu.pl

Abstract. The fours model designed to support competence management are presented: the model of knowledge processing in Open and Distance Learning, the motivation model, the model of developing a curriculum oriented on competence acquisition, the model of competence development in an intangible production network. All the models are integrated through common parameters and criterions and are adapted to Digital Economy context. In the paper the approach to models integration is proposed. The models are designed to be implemented in competences acquiring process for knowledge repository development in the distance learning network (DLN). The DLN's aim is to 'produce' the competence required on the job market.

Keywords: Competence management, Open and Distance Learning, ontology.

1 Introduction

The research in the area of knowledge management on the basis of competences was so far developed in many domains. In industry [1,2] the research was oriented at preparing formal description methods and designing dedicated decision support systems limited to the specified companies and industry domains. In education [19] software and organisational infrastructure were prepared, supporting the concept of Long Life Learning (LLL) and allowing combination of an individual development plan with the resources and activities enabling development of particular competences. These activities were mainly oriented on processing meta-structures of competence description. In business [3] management of competences was considered in the context of managing human resources (HR) and knowledge recourses [5,13]. In this perspective, an important research problem is the method of determining the competence resource possessed by a given person.

All these approaches are characterised by using results of cognitive science and using computer structures of ontology for modelling knowledge and competences. However, they do not provide computer tools for managing the content of knowledge within competences, depending on the goals of education and the syllabuses. Moreover, they do not take into account the phenomenon of knowledge growth in the repository as a result of the motivation of participants of the learning–teaching

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 604–613, 2012.

process [6,12,13]. In the discussed approaches standards were used for describing the area and category of competences (HR-XML, IEEE RCD), what is not sufficient to present the cognitive process as a set of intelligent operations related to representation and mastering of different types of knowledge in a computer environment. In the approaches presented above ontology was developed in an intuitive way. In higher education it is essential to consider the general, hierarchical layout of ontology: level of specialisation, course, topic. The ontology of the specialisation can be considered as a combination of top-level type ontologies from different domains [9]. The most important limitation of the proposed models is the lack of computational methods which would allow to assess the efficiency of the process of acquiring competences on the basis of proposed ontologies.

The research of the author is different from the works of others in the same domain in that the process of competence acquisition was interpreted as a production process, which consists of a defined sequence of intelligent operations realised in a computer environment [9,10,11]. The complexity of this production process is best researched on the example of an educational organisation where competences acquisition by each student has to be considered within the frames of a multi-level system of managing the entire organisation.

With traditional forms of teaching the process of competence acquisition was considered as a deterministic process. In the Open and Distance Learning (ODL) conditions we are dealing with the stochastic nature of this process [11]: unplanned changes in syllabuses resulting from new market requirements, unplanned education process in asynchronous mode, large discrepancy in time or performing tasks by the students and the teachers. The stochastic nature of the process of competence acquisition in the educational organisation was addressed through the use of modelling based on the Queuing Systems Theory.

In the proposed new approach to modelling and analysis of computer knowledge management systems, the object of research is a computer system supporting competence management in ODL conditions. The goal of the research is developing a model of a production network ensuring acquisition of competences required on the job market. Overcoming the explained drawbacks of other solutions requires developing a computer system that will be based on formalisation of the following issues:

1. Modelling the process of acquiring required competences by each student has to be considered within the entire system (multi-level, open) of managing an educational organisation functioning as an organisation based on knowledge,
2. Multi-level representation of ontology of different types of domain knowledge should be interpreted as a structure for storing didactic materials in the repository,
3. Planning and monitoring the state of the repository as an instrument of managing cooperation of participants of the learning–teaching process based on the model of cooperation with consideration of appropriate motivations,
4. Interpretation of the process of learning–teaching in asynchronous mode as a model of a queuing network.

The paper’s goal is to present the approach to the competence management, which support all (1-4) issues. The proposed approach to the competence management is a result of author’s research in the area of knowledge and competence management for several years. The models are spread around a number of papers. This paper collects the models and discussed the issue of models integration. Moreover, the paper includes a number of references to author's work related to the competence management issue. The proposed models can be easily adapted to the in Digital Economy context. As motivation for research work related to the topic of competence management the author used the following academic and professional experience:

1. Professional work as a computer scientist and an academic teacher
   Conducting lectures, practical classes and scientific circles was the basis for making observations on the difference between the presentation and reception of information and knowledge and the acquisition of competence. The author watched as over 10 years, through the emergence of new technologies, it becomes possible to recognize and formalize these differences. The results of observations can be verified by studying the results of cognitive science, which allowed the use of a process approach for describing the cognitive process as the process of acquisition of competences in the form of predefined types of knowledge in a particular topic.
2. Participation in the e-Quality project: Quality implementation in open and distance learning in a multicultural European environment (2003-2006)
   The project was finances by the European Union program Socrates/Minerva. The following institutions participated in the project: European Universitary Pole of Montpellier and Languedoc-Roussillon (France, coordinator), University Montpellier 2 (France), Open University of Catalonia (Spain), University of Tampere (Finland), Szczecin University of Technology (Poland), University of Applied Sciences Valais (Switzerland), Lausanne University (Switzerland). The cooperation showed the need to implement clearly defined roles through identification of the related activities and artefacts. Without clear definition of roles it is impossible to realise the process approach. It is also difficult to define the scenario of cooperation for the actors. The formalization of this approach is the proposed model of cooperation of students and teachers in preparing the teaching materials in an Open and Distance Learning (ODL) environment in the form of a repository. The operation of this model was checked by organizing and conducting training sessions in the e-Quality project.
3. Working in the research group
   Knowledge sharing has allowed the integration of the multi-faceted nature of the discussed subject and a number of scientific methods to form models compatible with the requirements of IT. The resulting synergy has allowed the development of a common concept of an Intelligent Open Learning Systems [11].

2 The Competence Management Issue

At the beginning we should analysis if the competence management in education or other intelligent process related to the in Digital Economy is possible. The task of any management system is to track state changes and improve the operation of the management's object. According to the theory of management, you can only manage the process of object behaviour. In the case of the competence management the behaviour of process, which can be management, is acquiring different types of knowledge. In proposed approach the competence consists of an appropriate combination of the theoretical knowledge, procedural knowledge, and project knowledge [9,10]. The behaviour of the object leads to a change in his condition. In the context of competence management process the management process can change the state of different types of knowledge in the curriculum, the state of knowledge contained in the current competence of the student, and the state of different types of knowledge contained in the repository. Changing the status of an object can be made by assessing the competence ontology graph [7]. In the case of competence management systems change in state can be determined by measuring the content knowledge based on the ontological structure [11]. Preliminary analysis of the system allowed to evaluate the possibility of creating this system (Table 1) by checking the fulfilment of the axioms of management.

Table 1. The axioms of management discussion in context of the competence management system

Axioms of management	Competence management system context
The object is suitable for observations and measurements	The content of knowledge can be represented and analysed on an ontological level [8]
At the interval of observation object can change its state	The competence is changed during the learning-teaching process
The predetermined target defined expected object's state	Goal is defined in the qualifications required by the market
There are alternative ways to influence the behaviour of an object	There are various models of distance education network utilization [11] focused on both the student-teacher relationship and student-student (social networks)
There is a pre-defined criterion of management efficiency	The criterion determines the degree of matching acquired competences to market requirements
There are resources for the execution of the decision	Resources include, among others: the teacher, LMS / LCMS information systems [11]

In discussed context the competence management is a process of tracking changes in content of knowledge related to the competences. The competences acquisition is guaranteed by learning–teaching process in the context of specific educational programs [11]. Each management system is based on the model, which has the task of mapping a process. In the case of the system of knowledge management model is interpreted as an ontological model. The use of ontologies allows the mapping process of competence acquisition in terms of the learning–teaching process.

Analysing the nature of the acquisition of competence management model, we can classify it as a management model with incomplete and uncertain information about the current resources of the knowledge contained in competence. The global criterion can assume the degree of adjustment of acquired competences to market requirements taking under consideration the time limits and curriculum.

3 Proposed Models for Competence Management

3.1 Approach Description

The competence management process was analysed based on the distance learning network structure [4,16]. The distance learning network is a complex computer system dedicated to competence-based education [17]. The most important of cognitive rules related to the process of learning–teaching is the networked nature of knowledge structure. The conducted analysis allowed to develop a model of network functioning, based on the concept of CBS (Community-Built System) [14]. Analysis of social characteristics of the discussed problem allowed for formulating a motivation model based on [6,12].

The used methodology of soft system analysis allows to determine semantics of relations between actors and to develop a detailed model of behaviour of the actors when creating a new form of didactic material in the form of a repository [17]. The model includes the active position of each actor. Verification of the proposed system can be performed on the basis of a simulation model of the process of competence acquisition by students.

At the next step intangible production network was designed for the needs of acquiring competences as a result of knowledge repository development in distance learning network [11,16]. For the purpose of describing the model the mathematical apparatus of queuing networks was chosen, where each workplace is a queuing system with appropriate parameters. Such interpretation allows the use of approaches relevant in other industries and allows to simulate the process of knowledge management. The model presents the learning process in the form of a network production process, which consists of a set of workplaces. For each of the workplaces tools, operations and requirements regarding the half-product/product were determined. The presented model refers to technology in which knowledge is the material and competence the final product. Verification of the presented model was based on a simulation experiment described in [17].

Not all the models related to competence management can be realised by strict analytical methods. The starting point for analysis and determination of the structure of competence from the computer point of view is the cognitive approach. Such an approach allows to distinguish stages of competence acquisition for which appropriate methods and algorithms have already been developed [11]. The most important ones are methods for describing the knowledge included in competences that are based on ontologies of different levels. It is also important to interpret the process of acquiring guaranteed competences by each student as a Markov process [10]. This allows for determining the intensity of servicing all students at each workplace in the production network. The passing of each student through all workplaces (which signifies appropriate intelligent operations have been performed by the student) is the basis for acquiring guaranteed competences.

The analysis made in [14] leading to the development of a model of educational situation in which the growth of knowledge and development of the repository is realized through cooperation between different actors. The result of this analysis is the cooperation model built according to the CBS concept. The cooperation model of each participant with another one is interpreted as a Markov process, which allows to determine basic characteristics of a production network system [10].

A supplementation to the discussed model is a dynamic model of knowledge growth in the repository at a specified interval of time with a specified time unit. The process of repository development is considered as a stochastic process. Randomness results from the cognitive nature of the process of student's knowledge growth, as well as from organizational reasons causing significant deflection from the planned timetable for the learning–teaching process.

3.2 Models for Competence Management

Model of knowledge processing in Open and Distance Learning
(The formal definitions of presented model can be found in the paper [11]).
Structure of the model:

- A set of participants of the learning–teaching process along with a description of their roles and responsibilities.
- The content and type of knowledge contained in a particular subject (course).

Goal of the model:

- Integration of different kinds of knowledge and the different manner of its presentation in a single model based on the ontology approach, with further fragmentation of this model into a set of knowledge portions in the form of Learning Objects.

Set of tasks:

- Developing a method for accumulating various types and fragments of teaching materials, which reflect the declarative (theoretical), procedural and design knowledge through the use of a common repository,
- Developing a method for decomposing the integrated knowledge model into knowledge portions in accordance with the principles of teaching and cognitive science,
- Developing a method for learning process personalization in accordance with student‘s individual preferences by selecting the appropriate method of knowledge representation and the choosing between deductive and inductive method of learning.

Motivation model
(The formal definitions of presented model can be found in the paper [12,13]).
Structure of the model:

- The student's competence acquisition process as a chain of intelligent operations for the performing tasks with certain conditions for completion,
- The process of supporting the students' work by the teacher through collaboration and tasks assessment,
- The process of changing the degree of repository completeness, depending on the effectiveness of student's performance with the respect to the tasks.

Goal of the model:

Modelling mutual influence of the described processes on the growth of knowledge in the repository within a defined time frame, depending on: (a) the teacher's choice of strategy based on his/her research and teaching interests and time constraints, (b) the degree of ambition and proficiency level of each student, reflected in the selection and execution of tasks, and his/her time constraints.

- The criterion is the following: the rate of filling in of the repository and the maximum growth of knowledge during the process of teaching each student individually and all students together.

Set of tasks:

- Developing the student's award function on the set of tasks that ensure the fastest repository growth within the specified time interval,
- Developing the student's preference function on the set of tasks through the choice of acceptable meanings of the effectiveness of task's solution,
- Calculation of the performance efficiency parameters of the repository filling-in process management system.

Model of developing a curriculum oriented on competence acquisition

(The formal definitions of presented model can be found in the paper [7]).

Structure of the model:

- A set of data characterizing the competence labour market demand together with a description of the related jobs (competencies required),
- A set of data characterizing the state of the market of educational offers described through the specializations, interpreted in the model as a set of guaranteed competences, programs and learning cycles.

Goal of the model:

- Use of an object model for describing functional relationships between the above-defined data sets and procedures for their processing, allowing to determine the balance between the amount of the competences required by the labor market and the amount of educational offers.

Set of tasks:

- Determining the time required for matching the current curriculum with a known intensity of the emergence of new technologies or new competences required on the market.
- Developing a functional diagram of specialization life cycle.

Model of competence development in an intangible production network

(The formal definitions of presented model can be found in the paper [9,10]).

Structure of the model:

- A set and sequence of intelligence operations that consist for the entire process of competence acquisition by the student within a particular specialization,
- Operation of acquisition of each type of knowledge related to a specific competence,
- Characteristics of the flow (intensity) of arrival/ beginning of performance of each operation by the student.

Goal of the model:

- Finding the structure and parameters of the open production network, which interprets the student's competence acquisition process.

Set of tasks:

Definition of the open production network performance parameters such as: the average number of students who are in the process of learning; the average number of students who have successfully completed the learning process; the average time spent by a student in the network, the average waiting time for access to the teaching resources; the speed of filling in the repository of knowledge in the learning process.

3.3 Generalization Model

The figure 1 shows the integration of models in common process of knowledge processing in Open and Distance Learning (ODL). The utilization of the ontology approach allows to: (i) processing knowledge portions in the form of Learning Objects [20], (ii) develop different type of knowledge in knowledge repository, (iii) analyses the quantity characteristics of competence based on the competence set approach [18]. The motivation model supports cooperation at the level of creating content and is focused on the task of filling the knowledge repository with high quality didactic material.

The model of developing a curriculum oriented on competence acquisition formalized the decision about curriculum modification, which usually takes place at the knowledge level, mainly with consideration of individual academic staff competences and qualifications. The educational organization, which works according to the ODL objectives, is obligated to conduct continuous analysis of competences required by the market. The educational organization guarantees competences acquired through its curriculum and learning-teaching process organization.

The final stage of knowledge processing in ODL is the model of competence development in an intangible production network. The competence development process occurs during the learning-teaching process based on e-learning. The production process is maintained in the educational organization according to the ODL concept. Intangible production is an advanced manufacturing process performed on the information level, where input materials, semi products and final products are in a digital form. The production network consists of nodes, each of which performs processing of information and knowledge through collaboration with other nodes. The model focuses on the kind of intangible production where the production process utilizes different types of knowledge and competence is the final product.

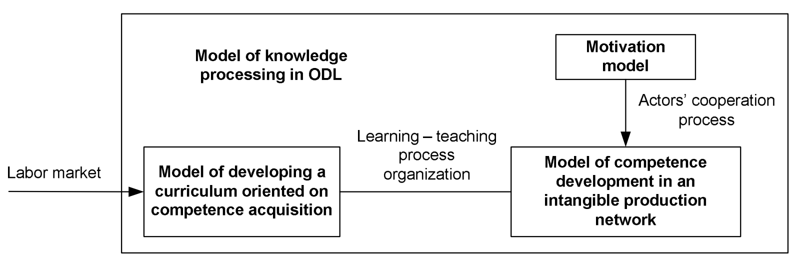


Fig. 1. Integration of models in competence management system

In order to make the communication between model more sufficient the Competence Object Library (COL) is proposed [18]. The reusable Competence Object Library (COL) integrates objective data structures complying with TENCompetence Domain Model (TCDM) with method for fuzzy competence sets modelling and method of competence expansion cost analysis proposed in [18].

4 Conclusion

The added value to the theory of distance learning computer systems is the development of a conceptual model of functioning of the computer system aimed at competence acquisition, which consists of separate models (including): motivation and cooperation. This allows for placing the proposed computer system within the entire multi-level open system of educational organisation management.

The discussed models are important elements of competence-based architecture (CBA) developed by the author. The CBA adapted to knowledge logistics in the project-oriented organization, basing on the fuzzy competence set theory, is presented in [15]. The competence-based approach is focused on the computational aspect of knowledge management. The CBA can be applied in different types of knowledge-based organizations and become important elements of Digital Economy environment.

References

1. Berio, G., Harzallah, M.: Towards an integrating architecture for competence management. Computers in Industry 58(2), 199–209 (2007)
2. Boucher, X., Bonjour, E., Grabot, B.: Formalisation and use of competencies for industrial performance optimisation: A survey. Computers in Industry 58(2), 98–117 (2007)
3. Enders, A.: Management Competence: Resource-Based Management and Plant Performance. Physica-Verlag HD (2004)
4. Mulvenna, M., Zambonelli, F., Curran, K., Nugent, C.: Knowledge Networks. In: Stavrakakis, I., Smirnov, M. (eds.) WAC 2005. LNCS, vol. 3854, pp. 99–114. Springer, Heidelberg (2006)
5. Nemetz, M.: A Meta-Model for Intellectual Capital Reporting. In: Reimer, U., Karagiannis, D. (eds.) PAKM 2006. LNCS (LNAI), vol. 4333, pp. 213–223. Springer, Heidelberg (2006)
6. Kusztina, E., Zaikin, O., Tadeusiewicz, R.: The research behavior/attitude support model in open learning systems. Bulletin of the Polish Academy of Sciences: Technical Sciences 58(4), 705–711 (2010)
7. Kusztina, E., Zaikin, O., Różewski, P., Małachowski, B.: Cost estimation algorithm and decision-making model for curriculum modification in educational organization. European Journal of Operational Research 197(2), 752–763 (2009)
8. Kushtina, E., Różewski, P., Zaikin, O.: Extended Ontological Model for Distance Learning Purpose. In: Reimer, U., Karagiannis, D. (eds.) PAKM 2006. LNCS (LNAI), vol. 4333, pp. 155–165. Springer, Heidelberg (2006)
9. Różewski, P.: Model of Intangible Production Network for Competence Development. In: Xiong, H., Lee, W.B. (eds.) KSEM 2011. LNCS (LNAI), vol. 7091, pp. 246–256. Springer, Heidelberg (2011)

10. Różewski, P.: Concept of intangible production network system for competence development in open and distance learning. Management and Production Engineering Review 2(3), 52–62 (2011)
11. Różewski, P., Kusztina, E., Tadeusiewicz, R., Zaikin, O.: Intelligent Open Learning Systems: Concepts, models and algorithms. ISRL, vol. 22. Springer, Heidelberg (2011)
12. Różewski, P., Kusztina, E.: Motivation Model in Community-Built System. In: Pardede, E. (ed.) Community-Built Database: Research and Development, pp. 183–205. Springer, Heidelberg (2011)
13. Różewski, P., Zaikin, O., Kusztina, E., Tadeusiewicz, R.: The Motivation Model for the Intellectual Capital Increasing in the Knowledge-Base Organization. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 47–56. Springer, Heidelberg (2011)
14. Różewski, P.: Model of Community-Build System for Knowledge Development. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part II. LNCS, vol. 6923, pp. 50–59. Springer, Heidelberg (2011)
15. Różewski, P., Małachowski, B.: Competence-Based Architecture for Knowledge Logistics in Project-Oriented Organization. In: O'Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS (LNAI), vol. 6682, pp. 630–639. Springer, Heidelberg (2011)
16. Różewski, P.: A Method of Social Collaboration and Knowledge Sharing Acceleration for e-Learning System: The Distance Learning Network Scenario. In: Bi, Y., Williams, M.-A. (eds.) KSEM 2010. LNCS (LNAI), vol. 6291, pp. 148–159. Springer, Heidelberg (2010)
17. Przemysław, R.z., Magdalena, C.: Model of a Collaboration Environment for Knowledge Management in Competence-Based Learning. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS (LNAI), vol. 5796, pp. 333–344. Springer, Heidelberg (2009)
18. Różewski, P., Małachowski, B.: Competence Management in Knowledge-Based Organisation: Case Study Based on Higher Education Organisation. In: Karagiannis, D., Jin, Z. (eds.) KSEM 2009. LNCS, vol. 5914, pp. 358–369. Springer, Heidelberg (2009)
19. TENCompetence - Building the European Network for Lifelong Competence Development, EU IST-TEL project (2005-2009), `http://www.tencompetence.org/`
20. Zaikin, O., Kushtina, E., Różewski, P.: Model and algorithm of the conceptual scheme formation for knowledge domain in distance learning. European Journal of Operational Research 175(3), 1379–1399 (2006)

Materials Management in Recovery Network: Integration of Distributed Data Sources

Paulina Golinska

Poznan University of Technology, Strzelecka 11, 60-965 Poznan, Poland
paulina.golinska@put.poznan.pl

Abstract. Good organization of reverse logistics channels allows company to explore the additional value from extended life cycle of products. In order to be able to effectively plan flow of returns in recovery network a big number of data is needed. Moreover the exchange of information between the cooperating companies must be done in time efficient manner. The aim of this paper is to present a framework for information management among companies within recovery network. The recovery network for electronic equipment will be used as an example for theoretical considerations. The key factors and obstacles for coordination of recovery network will be identified. The requirements for integration platform functionality will be presented.

Keywords: supply chain, data exchange, recovery network, cooperation.

1 Introduction

Recovery networks are created by number of independent companies in order to collect products back from the customer. The aim is reprocessing of returns and their further redistribution to the market. The impetuses for development of recovery networks are legal obligations or potential economic benefits which might be obtained by catching the additional added value from prolong products life-cycles. Efficient recovery network can be an answer to problems that companies face, namely:

- the increasing cost of energy,
- the increasing cost of raw materials,
- the increasing cost of waste disposal.

Products recovery also helps to prolong the relationship with customers in the pos-sale phase by the application of the used products collection program like for example "give old-get cheaper new" actions. Efficient reverse materials management (re-supply) can be perceived as a new opportunity to lower the cost of the procurement. In order to gain the advantages of "two-way" economy, efficient information management is needed to balance both economic and ecological impacts [1].

Product recovery activities include used-product acquisition, reverse logistics, product disposition (sort, test, and grade), remanufacturing/repair, and remarketing [2].

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 614–621, 2012.

The potential value of recovery product must exceed the cost of recovery activities. The recovery network can be design based on the following approaches [3]:

- waste-stream cost minimization when compliance with legislative regulations is needed,
- profit maximization for market-driven cooperation.

In both cases success of the cooperation within the recovery network depends on access to appropriate data. Recovery networks (reverse supply chain) often involve many more independent players than forward supply chains. Moreover recovery networks are very often decentralized, what increases complexity of operations coordination among large number of actors (third-party reverse logistics providers, remanufacturers, recyclers ect.). There are big information asymmetries and incentive misalignment in the reverse supply chain [3]. The emphasis should be placed on a dynamic process management and coordination perspective.

In order to ensure continuity and efficiency of product flow from the client to the recovery network it is necessary to coordinate cooperation with partners who have appropriate delivery potential, who are able to ensure required reliability and flexibility and who are able to react fast. It results from data exchange within the process of common planning and scheduling of partners' operating activities in the recovery network.

The aim of this paper is to present a framework for information integration among companies within recovery network. The emphasis will be placed on the key success factors and obstacles for coordination of recovery network. The requirements for integration platform functionality will be presented.

The paper’s structure is as follows, the brief theoretical background is provided in Section 2. The highlights of time sensitive returns in computer industry are described in Section 3. Proposed model is presented in Section 4. Final conclusions are stated in Section 5.

2 Cooperation in Recovery Network

The aim of the recovery network is to plan and control how in the most profitably way collect the product returns from customers and recover them. The main problem for management of materials flow in recovery networks is uncertainty regarding timing of returns, quantity of returns, and their reusability ratio (so called material recovery rate MRR). The complexity of recovery network coordination increases with the comparative length of the reverse channel, volume, and differentiation of returns.

Important issue in materials management within recovery network is matching demand with product returns (re-supply).

Time is important issue in materials management in recovery network, because the longer it takes to retrieve a returned product, the lower the likelihood of economically viable recovery options [4]. The design of recovery network is connected with time sensitivity of returns. Responsive reverse supply chains are appropriate for products with high marginal value of time (MVT), whereas efficient reverse supply chains are appropriate for products with low MVT [5]. The marginal value of time is defined as the loss in value per unit of time spent awaiting completion of the recovery process [5]. Responsive recovery network aims to minimize the lead time. Efficient recovery network is designed to achieve the minimal unit cost of reverse logistics operations (including e.g. collection, transport, reprocessing). The delays in lead times in recovery network might cause situations when a value of returns will erode because of unwanted delays in collection of goods, their delivery and disposition (choice of recovery option). When the emphasis is placed on the cost efficiency, then the returns supply chain should be designed to centralize the disposition activities. When responsiveness is the goal, then a decentralized disposition activities are needed in order to minimize time delays in reprocessing returns. Early diagnosis of product condition maximizes asset recovery by fast-tracking returns on to their ultimate disposition and minimizing the delay cost [5].

In order to facilitate coordination and balance returns with demands it is important that timely and accurate information is available [6]. This can be achieved through investing in advanced information systems that track the location and condition of products in the market, for example through RFID tags (radio frequency identification). For more detailed on application of RFID in product recovery see [7].

3 Electronic Equipment Sector

Electronic equipment is a good example of products which need to be returned after the usage phase. Electronic equipment is also very often subject to the very time sensitive commercial returns. Electric and electronic equipment's waste (WEEE) is a big problem in number of well developed countries. The customers exchange computers, printers or servers more and more often. Since 2005, member states of the European Union have begun implementing the WEEE Directive, which requires manufacturers to provide the drop-off points for unwanted electronic devices. The manufacturers might reprocess the WEEE in number of ways. The most typical are: remanufacturing, recycling and redistribution.

The challenge is to reach by reprocessing the “win-win” situation, where both environmental and economic goals are met. The useful life cycle might be prolonged by reselling refurbished equipment, remanufacturing components as spare parts or reclaimed raw materials.

The number of installed PCs worldwide has growth over 1 billion units, according to Gartner, Inc. The analysts estimate the worldwide installed base of PCs is growing just under 12 percent annually. At that pace, it will reach 2 billion units by early 2014.

Computer industry is especially interesting application domain for reverse logistics because of the following characteristics:

- mass production(big materials flow),
- short life cycles,
- potential for reprocessing.

Many ‘end-of-life’ products are still in good working condition and may therefore find useful application. On the other hand, quick depreciation puts this option under significant time pressure [5].

The environmental friendly practices in computer industry include mainly:

- extension of the products useful life when possible through a refurbishing process,
- proper disposal of units that cannot be recovered.
- utilization of useful components and materials from recyclable products.

Electronic equipment often contains heavy metals and other hazardous substances, and must be refurbished or recycled properly. On the other side used IT equipment can be a source of valuable resources. Most of the top producers began pursuing environmentally friendly disposal policies long before they were mandated by government regulations. Nowadays the leader in reprocessing practices is HP. The company offers for its customers recycling and reuse programs. HP has aimed to recycle 900,000 tones of electronic products and supplies by the end of 2010 (since 1987). In reality 884,000 tones of electronic products and supplies has been recycled. HP also uses a network of vendors (recovery service providers) to process, resell and recycle return products. The company issues Global Reuse and Recycling Standards which define conditions and requirements for storing, handling and processing returned equipment in ways that prevent from the leak of harmful substances. Nowadays company has in the pool in recovery network about 500 recycling providers’ locations around the world. Institutions providing reuse services for HP products must ensure that those operations occur in on time and secure. The time issue is highlighted by the network configuration. The company tries to limit the storage of goods for recycling ore reuse must not be stored for more than 90 days.

Another example of recovery network configuration is IBM. IBM’s asset recovery assumes priority on reselling equipment as a whole. The products are sold through IBM’s sales network as certified remanufactured equipment. The business partners and brokers are also involved in these operations. For the PC sector, IBM uses a different channel. The products are tested by Global Assets Recovery Services (GARS) and then are auctioned off in large batches to brokers. The valuable components are delivered as spare parts to IBM’s own service division. In addition, GARS also sells recovered components to external brokers. The remaining equipment is recycled.

A professional asset recovery program provides a good opportunity to find value in older equipment and to enhance the organization’s reputation for environmental friendly institution. Professional recovery companies have specialized skills: logistics (inventory control, transport, storage, etc.), data wiping, equipment refurbishment, resale and environmentally responsible recycling [8]. It is very important to identify at early stage if the product has never been used, lightly used, or heavily used. Given this information, the product can be more effectively channeled to the desired processing facility, saving time, and improving the added value of product recovery. The example of the recovery network in computer industry is presented in figure 2.

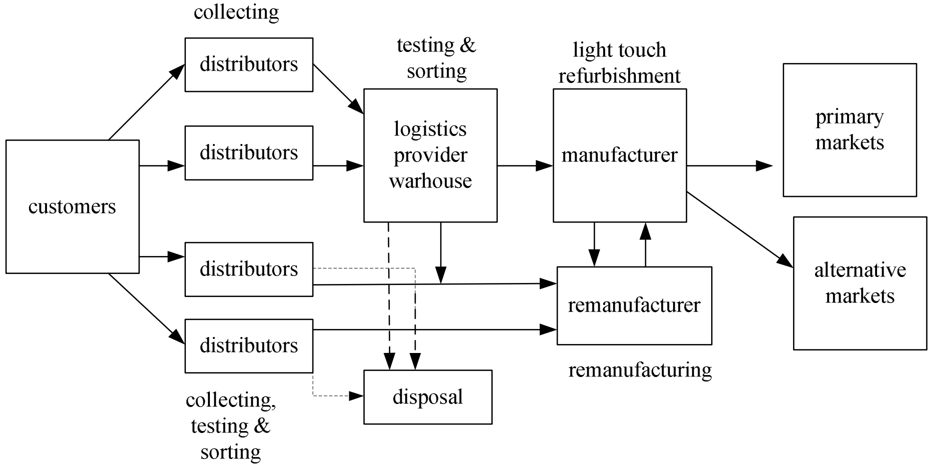


Fig. 1. The recovery network in the computer industry

4 Requirements for Integration Platform

Product returns have recoverable value and are potential sources of revenue; especially when the time between product return and resale is short [4]. Recovery networks are often fragmented, with numerous entities focused on local efficiencies and economies of scale. Managers lack clear end-to-end visibility of processes. Masses of data may be captured but remain unprocessed, unconnected, and scattered throughout the organization [9]. Decision makers need information on:

- reprocessing lead time,
- volume of returned product flows,
- possibilities for grouping products families (commonality),
- locations of potential collection points (distribution points),
- waiting time for returns transportation,
- volume of returns inventories in distributed locations,
- value erosion patterns (distribution of marginal value of time for particular products parts),
- quality distribution (usage rate- statistics on stochastic patterns of recovery rate),
- availability of recovery services (updated information on reprocessing capacities and queuing time for particular recovery services).

Very often such data is not easily available. Integration of materials management within recovery networks requires information that is generated and stored in the databases and information systems that are dispersed amongst the whole supply chain. The technical issues related to the design of the integration platform/interface are not covered by the scope of this paper. The aim here is only to present the scope of information needed by the planning of efficient and environmental friendly materials management activities. The integration platform allows executing following procedure (compare fig. 2):

1. Identify the volume of returns for particular period (forecasting),
2. Identify collection points,
3. Verify the quality of returns (disposition options)
4. Divide the returns flow (disposal or manufacturer or recovery service provider e.g. remanufacturer),
5. Assign the products families (based on product type and the technical state of product),
6. Estimate the reprocessing lead times for particular products,
7. Allocate the returns to recovery service provider with available capacities and shortest queuing time,
8. Assign the transportation routes,
9. Examine the quality of recovered products
10. Enter the reprocessed products to the inventory,
11. Plan redistribution (re-delivery schedule).

In order to perform the above procedure dispersed data sources need to be integrated. Taking in consideration the characteristics of recovery network the following sources should be linked via web-based platform:

- Recovery Service Provider's (RSP's) Information Systems – should provide information regarding life cycle inventory, lead times for recovery processes, capacities for recovery facilities, disassembly and re-assembly.
- Manufacturer ERP/PPC- provides data on procurement, production and sales, especially information about bill of materials (BOM), production orders, production capacities, disassembly BOM (if applicable, DBOM could be also obtained from external data sources),
- Distribution Systems- provide data regarding detailed location of collection points and amounts of sales (needed for returns flows forecast and the design of recovery network),
- External Life Cycle Assessment databases (LCA DB) - provide the estimates on product life cycle assessment based on products classification as well as data needed for returns forecasts and assessment of returns quality.

Figure 3 presents the process layer and information layer in recovery network. The integration of dispersed data sources between recovery network participants is crucial in order to bridge the information gap between the middle-of–life phase and end-of-life repair and remanufacturing processes. The proposed approach simplifies the decision making process and helps with optimizing the materials management based on complete information from post-sale phases. The integration of the above mentioned data sources allow the goal of reduction of resource allocation and emissions by maximization of returns and minimization of raw materials and new components usage to be reached.

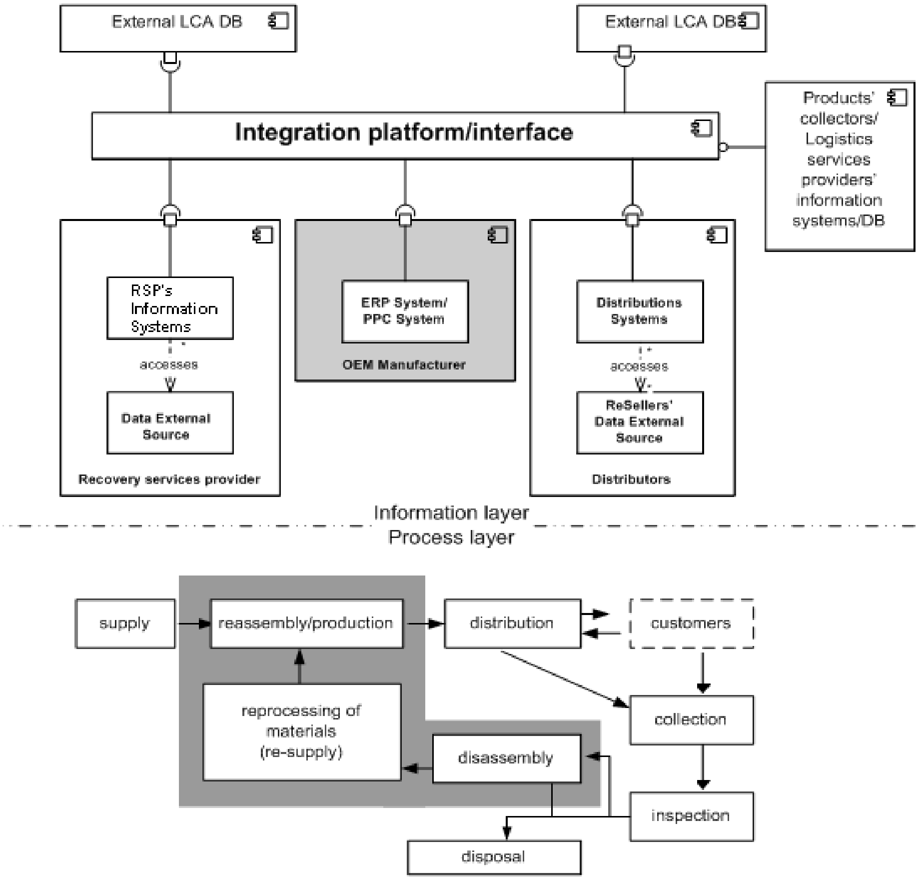


Fig. 2. Integration platform model

5 Conclusions

The shift from traditional manufacturing to sustainable business practices is still a challenge. However the number of companies that are involved in product recovery operations is growing. This paper presents the areas that are crucial for improvement of the materials management within recovery network.

The main challenge that requires additional research is still the problem of collection and utilization of dispersed data sources. As pointed out in the text, materials management in the recovery network needs very complex data originating from systems of different structure. It still can be observed in industry practice that post sale parts of the supply chain act as black holes for planning activities. An integrated information flow regarding the nature, quantity and quality of returns flows is a must. The proposed approach of integration identified the scope of information that is crucial for efficient material management between product life cycle phases.

The main limitation of the proposed approach is the lack of existing cost-efficient integration platforms that can synchronize the information exchange between the parties that are involved in the recovery network cooperation. Further studies should focus on the identification of the integration platform technical requirements. The issue related to increasing trust and safety for information exchange between entities involved in recovery network should be examined.

References

[1] Golinska, P.: From Traditional Non- Sustainable Production to Closed Loop Manufacturing: Challenges for Materials Management Based on PPC and EMIS Integration. In: Teuteberg, F., Marx Gomez, C. (eds.) Corporate Environmental Management Information Systems: Advancements and Trends, Hershey, New York, pp. 106–120 (2010)

[2] Guide Jr., V.D.R., Van Wassenhove, L.N.: The reverse supply chain. Harvard Bus. Rev. 80(2), 25–26 (2002)

[3] Guide Jr., V.D.R., Van Wassenhove, L.N.: The Evolution of Closed-Loop Supply Chain Research. INFORMS 57(1), 10–18 (2009)

[4] Ketzenberg, M., Van der Laan, E., Teunter, R.H.: Value of information in closed loop supply chains. Production Oper. Management 15, 393–406 (2006)

[5] Blackburn, J., Guide, D., Souza, J., Van Wassenhove, L.N.: Reverse supply chains for commercial returns. California Management Review 46(2), 6–23 (2004)

[6] Kokkinaki, A., Zuidwijk, R., van Nunen, J., Dekker, R.: ICT enabling reverse logistics. In: Dekker, R., Inderfurth, K., Van Wassenhove, L., Fleischmann, M. (eds.) Reverse Logistics, Quantitative Models for Closed-Loop Supply Chains, p. 381. Springer, Heidelberg (2004)

[7] Karaer, O., Lee, H.L.: Managing the Reverse Channel with RFID-Enabled Negative Demand Information 16(5), 625–645 (2007)

[8] Golińska, P., Kawa, A.: Recovery network arrangements -WEEE case. In: Golinska, P., Fertsch, M., Marx Gomez, J. (eds.) Information Technologies in Enviromental Engineering- Trends and Challenges, pp. 579–593. Springer, Heidelberg (2011)

[9] Guide Jr., V.D.R., Muyldermans, L., Van Wassenhove, L.N.: Hewlett-Packard Company Unlocks the Value Potential from Time-Sensitive Returns. Interfaces 35(4), 281–293

Optimization of Supply Chain via Reduction of Complaints Ratio

Marcin Anholcer and Arkadiusz Kawa

Poznań University of Economics,
al. Niepodległości 10, 61-875 Poznań, Poland
{m.anholcer,arkadiusz.kawa}@ue.poznan.pl

Abstract. The food supply chain is a very demanding and dynamic sector of economy. The goods that flow through the supply chain are exposed to damage. The authors have developed a model which optimizes the cost of transport in the network. It is based on graph theory, and, in particular, the Generalized Flow Problem has been used. The main contribution of this paper is the implementation of this model in a simulation environment. It has allowed to research how defect ratios influence the structure of an optimal configuration of a supply chain.

Keywords: supply chain, distribution, complaints, network model, simulation experiments.

1 Introduction

Safety is a very important part of logistics. Entrusting a logistics company with their commodity, clients want to be sure that it will go to the customer not only to the right place, at the right time and quantity, but also in good condition. What is meant by "good condition" is delivering items without damage, quality deterioration, change in their status, weight, capacity. It is not only about the flow of goods within a single firm, but also the flow between various companies. The movement of goods (raw materials, intermediate products, finished products, etc.) through networks of enterprises, which are suppliers and customers to each other, is called the supply chain. This flow is associated with a single product or a group of products and also applies to deliveries to the final customer.

An important component of the supply chain is transport between the different entities in the network. Every good must, in fact, be moved from one place to another. It should be noted that transport is one of the most costly logistics processes and most susceptible to damage.

The reason for damage to cargo in transport may be inappropriate protection of the goods, lack of equipment, inadequate driver qualifications, maladjustment or poor condition of the means of transport. In addition to the technical premises contributing to the damage in transport, economic reasons can be distinguished, too. A serious problem may lie in more and more common price pressure, which forces supply chain

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 622–628, 2012.

leaders into constant efforts to lower prices, even when the competitive strategy is primarily focused on such characteristics as quality or delivery time. Globalization and internationalization of enterprises have also contributed to the fact that a large number of European and American companies produce or commission production in the Asian market, which greatly lengthens the transport process, and this contributes to a greater risk of damage.

Damages result from road collisions, the product getting wet, dry or crushed. Of course, attempts to take specific actions to reduce the risk of damage may be made, but they are not always fully effective. The possible actions include: introduction of appropriate procedures and instructions, monitoring vehicle work (GPS), driver training, educating managers to control the execution of specific tasks.

Economy can be divided into sectors in which goods are particularly vulnerable to damage. These include glass-making industry, electronic and food industry. The last one, in which the so-called food supply chain appears, is particularly interesting.

The food supply chain is an integrated process where raw materials are acquired, converted into products and then delivered to the consumer. The chain is linked with feed forward flow of materials and the feedback flow of information [4]. Tijskens et al. conclude that the main fact that differentiates food supply chains form other supply chains is a continuous change in quality from the time the raw materials leave the grower to the time the food reaches the consumer [8].

It is worth to emphasize that food production and distribution is a big business. The U.S. food and fiber system is the nation's largest manufacturing sub-sector, accounting for over 12% of GDP and 17% of national employment. And food wholesaling and distribution represents a large chunk of that - $372 billion, to be exact [2].

Because of these costs and damage in transit, food supply chain configuration will be needed. Merriam-Webster Dictionary and Online Thesaurus define configuration as "…something (as a figure, contour, pattern, or apparatus) that results from a particular arrangement of parts or components" [5]. In the case of system configuration (e.g. of an enterprises network) creating a given structure composed of particular elements (e.g. enterprises) is involved. In turn, the configuration of a supply chain may be characterized as the particular arrangement or permutation of the supply network's main elements, including the network structure of the various operations within the supply network and their integrating mechanisms, the flow of materials and information between and within key unit operations, the role, inter-relationships, governance between network partners, and the value structure of the product or service delivered [7].

The problem of the food supply chain is how to choose the right configuration, which will take into account the risk of damage and minimize transport costs. This paper aims to identify solutions to this problem. In order to do that, a specially developed model based on graph theory, and, in particular, the Generalized Flow Problem has been used. The model has been implemented in a simulation environment. Thus, the authors were able to study how changes in the parameters of the flow of goods through the supply chain can affect the structure of the network.

2 Distribution with Complaints – The Network Model

Let us consider the following supply network model of food. There are m supply points (factories) $S_1, S_2, \dots, S_m$, where supply point S_i can deliver s_i units good, p transshipment points (warehouses) $T_{m+1}, T_{m+2}, \dots, T_{m+p}$, where the capacity of transshipment point T_k is not bounded from above and n destination points (retailers) $D_{m+p+1}, D_{m+p+2}, \dots, D_{m+p+n}$, where the demand of point D_j (equal to the sum of the demands of final customers) equals to d_j. If we want to discuss the supply, destination and transshipment points together, we will refer to them as to the *agents* or *nodes* and denote with N_i ($i = 1, 2, \dots, m + p + n$). We assume that given good is transported from the supply points to the transshipment points and from the transshipment points to the destinations. Direct transportation from factories to retailers is not allowed. The example of such a food supply network is presented at fig. 1. The symbols are explained below.

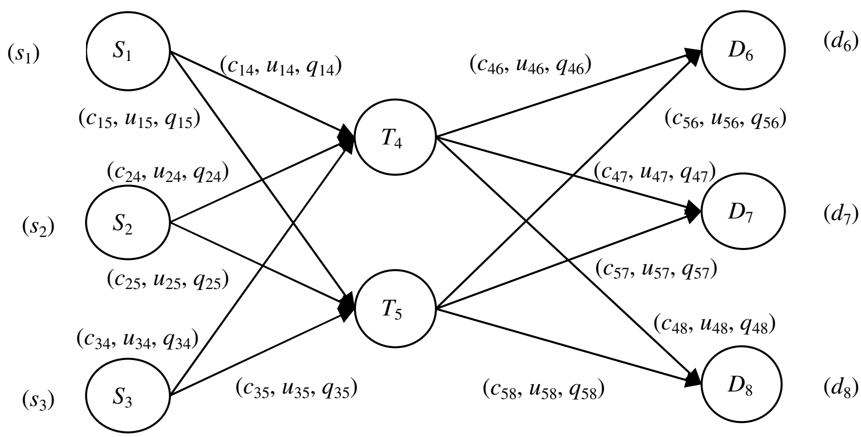


Fig. 1. Sample food network ($m = 3, p = 2, n = 3$)

We assume also that the complaints of final customers are possible. In such a case the retailer is obliged to exchange damaged product and send it back to the transshipment point and further to the factory. Also at the first stage of the distribution some defects may be found. Then, the damaged unit is sent back from the warehouse to the factory. We assume that the defect ratio for each pair of connected agents (N_i, N_j) is known and equals to r_{ij}. It means in particular, that if the retailer D_j obtains x_{ij} units of good from the warehouse T_i, then he is able to sell $(1 - r_{ij})x_{ij}$ units to the customers (the remaining $r_{ij}x_{ij}$ units are also sold, but they return and have to be exchanged). However, we can not modify the demands and warehouse capacities themselves, as the defect ratios for different pairs may not be simple combinations of some theoretical ratios attached to the agents. It follows from the fact that the defects are caused not only by the production process and storage, but also by the transportation system, that may be different for every pair of agents.

Observe that the transportation cost depends on the amount of good that leaves given node, not on the amount that comes to it. Thus, if we denote the unit transportation cost between agents N_i and N_j by c_{ij}, then the total cost will be equal to $c_{ij}x_{ij}$. Let us define the reliability ratio $q_{ij} = 1 - r_{ij}$ and the maximum capacity of every connection u_{ij}. Then the optimization model that lets us to minimize the total cost of transportation is given as follows.

$$\min f(x) = \sum_{i=1}^{m} \sum_{j=m+1}^{m+p} c_{ij}x_{ij} + \sum_{i=m+1}^{m+p} \sum_{j=m+p+1}^{m+p+n} c_{ij}x_{ij},$$

$$s.t.$$

$$(1) \sum_{j=m+1}^{m+p} x_{ij} \leq s_i, \quad i = 1,2,\dots,m,$$

$$(2) \sum_{j=1}^{m} q_{ji}x_{ji} = \sum_{j=m+p+1}^{m+p+n} x_{ij}, \quad i = m+1, m+2, \dots, m+p,$$

$$(3) \sum_{j=m+1}^{m+p} q_{ji}x_{ji} \geq d_i, \quad i = m+p+1, m+p+2, \dots, m+p+n,$$

$$(4) 0 \leq x_{ij} \leq u_{ij}, \quad i = 1,2,\dots,m, \quad j = m+1, m+2, \dots, m+p,$$

$$(5) 0 \leq x_{ij} \leq u_{ij}, \quad k = m+1, m+2, \dots, m+p,$$
$$j = m+p+1, m+p+2, \dots, m+p+n.$$

The objective is to minimize the total transportation cost. Constraints (1) are supply constraints guaranteeing that the total amount of good delivered by any factory to all the warehouses does not exceed the factory's supply. Constraints (2) are typical mass balance constraints (amount of good incoming to every warehouse is equal to the amount of good leaving it). Constraints (3) are demand constraints ensuring that the demand request of every retailer will be satisfied. Constraints (4)-(5) guarantee in turn that the amount of good transported over any connection does not exceed its capacity.

The problem formulated as above is a variant of Generalized Flow Problem (GFP) [1,3] and given any instance of this problem we can solve it with Generalized Network Simplex Method [1]. The constraints (3) may be in fact rewritten in the form of equations (all the coefficient and variables are non-negative and we are minimizing the objective value, so at the optimal solution all the demand constraints will be binding). Constraints (1) may be not binding, however we can resolve this problem using artificial arcs of the form (S_i, S_i) with $q_{ii} = c_{ii} = 0$. The optimal value of each x_{ii} denotes the supply surplus of supply point S_i (at the node N_i).

In the optimal solution the arcs with positive flows form a spanning augmented forest (in particular it may be a spanning augmented tree). Here augmented means containing one additional arc. In this particular case all the additional arcs are loops (S_i, S_i). Two examples of such forests are give on fig. 2 and 3.

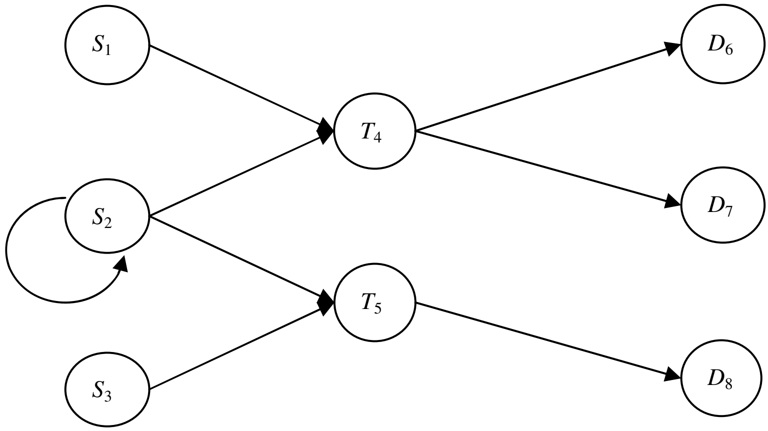


Fig. 2. Sample augmented forest ($t = 1$)

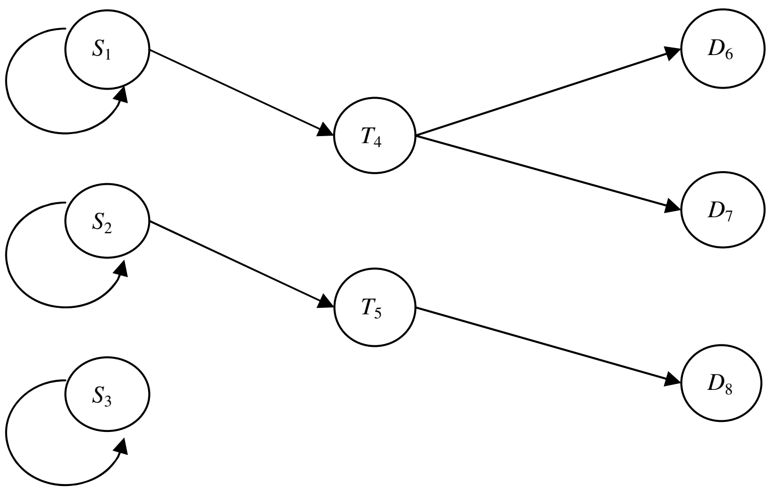


Fig. 3. Sample augmented forest ($t = 2$)

In the first case (fig. 2) all the agents form one augmented tree and the logistic network is quite complicated. In the second case (fig. 3) we have three trees: supplier S_1 collaborates only with the customers D_6 and D_7 using warehouse T_4, while supplier S_2 sends his products only two the customers D_7 and D_8, via the warehouse T_5. The third tree is formed only by the supplier S_3, which is excluded from the process. In both cases we are interested only in the "huge" trees, containing both suppliers and customers. Thus the number of trees in the figures 2 and 3 are $t = 1$ and $t = 2$, respectively.

As we can easily see, the second network is simpler and thus preferred. Each additional connection implies additional constant (transactional) costs and more organizational work. On the other hand, we can not assume the number of components – it follows as the outcome from the minimization of the total cost. This economic optimum does not need to guarantee maximization of the number of components. Our purpose was to check in which situations it is the case.

3 Simulation Experiments

The benefits of supply chain simulations are obvious. They have helped to understand the overall supply chain processes and characteristics by graphics and to capture system dynamics (using probability distribution). Moreover, thanks to simulation experiments, companies may perform powerful what-if analyses leading them to better planning decisions [6].

Using the model presented above we tested, how the defect ratios may influence the structure of the optimal configuration of supply chain. For that purpose we performed the simulation – we generated randomly some number of test problems with different values of parameters and check if the optimal structure changes.

Our assumptions were the following. Number of suppliers, warehouses and customers were 5, 10 and 100, respectively. The values of supply of every supplier were chosen independently uniformly at random from the interval <800, 1000>, the values of demand of every customer from the interval <10, 15>, the transportation costs on the connections between the suppliers and warehouses from the interval <5, 10> and the transportation costs on the connections between the warehouses and customers from the interval <10, 15>. We also assumed that the flows on all the connections are not bounded from above. We found the number of “huge” trees for various values of reliability ratio, chosen from three intervals representing typical values (0.9 and more) and two intervals that could appear in the real life problems less frequently. For every interval 250 random test problems were generated. The average numbers of “huge” trees in the optimal solutions are presented in table 1.

Table 1. Average number of trees in the optimal solution

$<q_{min}, q_{max}>$	<0.7, 0.8>	<0.8, 0.9>	<0.9, 0.93>	<0.93, 0.96>	<0.96, 0.99>
t	2.028	2.840	3.916	4.116	4.208

As we can see, the number of trees t increases with the possible values of reliability ratio. In other words, the lower is the complaint ratio, the lower is the complexity of the optimal network.

4 Conclusion and Future Work

The transport of goods in the food supply chain is exposed to damage. It causes complaints of customers and rising costs. The model developed and implemented in a simulation environment by the authors has allowed to check the influence of defect ratios on an optimal configuration of a supply chain. Moreover, the simulations carried out have permitted to evaluate the operating performance prior to the implementation of a system and enabled the comparison of various operational alternatives without interrupting the real system.

We proved that the increase of the reliability ratio improves the complexity of the transportation network. It means that if one is interested in decreasing the complexity of the supply chain, then he/she should encourage the suppliers to reduce the defect ratio.

Of course, this model is general and can be used not only within food supply chains. That is why our future work will focus on the adaptation of the model in other branches.

We are also interested in how the change of reliability ratio of chosen suppliers influences their situation, i.e. if the decrease of their defect rates may cause the decrease of complexity of the part of network they belong to.

Acknowledgements. The paper was written with financial support from the Foundation for Polish Science [Fundacja na rzecz Nauki Polskiej]. The research was also supported by the grant of the National Center of Science [Narodowe Centrum Nauki] nr 2011/01/D/HS4/03543.

References

1. Ahuja, R.K., Magnanti, T.L., Orlin, J.B.: Network Flows. Theory, Algorthms and Applications. Prentice-Hall (1993)
2. Cubitt, B.: Food Distribution – The Supply Chain Optimization Challenge, http://www.foodmanufacturing.com/
3. Glover, F., Klingman, D., Phillips, N.: Netform Modeling and Applications. Interfaces 20(4) (July-August 1990)
4. Jiang, Y., Zhao, L., Sun, S.: A Resilient Strategy for Meat-food Supply Chain Network Design. In: 2009 IEEE International Conference on Industrial Engineering and Engineering Management (2009)
5. Merriam-Webster Dictionary and Online Thesaurus, http://www.m-w.com
6. Penlope, T.F.: A system dynamics model for supply chain management in a resource constrained setting, http://dspace.mak.ac.ug/bitstream/123456789/629/3/tulinayo-fiona-penlope-cit-masters-report.pdf
7. Srai, J.S., Gregory, M.: A Supply Network Configuration Perspective on International Supply Chain Development. International Journal of Operations and Production Management 26(5) (2008)
8. Tijskens, L.M.M., Koster, A.C., Jonker, J.M.E.: Concepts of chain management and chain optimization. In: Food Process Modeling. Woodhead Publishing Ltd., Cambridge (2001)

Multi-Agent Systems in Distributed Computation

Maja Štula, Darko Stipaničev, and Ljiljana Šerić

Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, Department of Modeling and Intelligent Systems, University of Split, R. Boskovica 32, 21000 Split, Croatia
{maja.stula,dstip,ljijljana}@fesb.hr

Abstract. The paper investigates the possibility of multi-agent systems application in distributed computation illustrated by an example of generic distributed computation multi-agent system. The system is defined upon finite state machine architecture with agent internal and external behavior. For testing purpose, the model was implemented in Fuzzy Cognitive Map (FCM). FCM is qualitative modeling and simulation technique that is based on discrete distributed computation. Our proposed multi-agent system has shown expected characteristics so the conclusion is that the proposed generic model could be used as a foundation for building distributed computation multi-agent systems.

Keywords: Distributed computation, multi-agent system, agent behavior, fuzzy cognitive map.

1 Introduction

Distributed computation can be seen as a sequence of states and events that might have been produced by executing a computation. Multi-agent systems are suitable for distributed computation. Introducing multi-agent systems into distributed computation systems can facilitate implementation of distributed computation systems and also provide novel characteristics to existing distributed computation systems like more autonomy of distributed computation components. Distributed computation multi-agent system case studies exist but without formal model of the system [1, 2].

Distributed computation is common property of variety of different systems and this is emphasized in chapter two. Next chapter provides short multi-agent systems overview. Fourth chapter offers formal generic model of distributed computation multi-agent system. Distributed computation properties memory, communication and processing are defined through finite state machine with agent's internal and external behavior. Implementation of proposed model is presented in the next chapter with Fuzzy Cognitive Map built as a multi-agent system. Developed distributed computation multi-agent system shows appropriate behavior and generates expected results.

2 Distributed Computation

Distributed computation is any process conducted by multiple agents or entities that perform operations on information and together generate resulting information. It is

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 629–637, 2012.

present in many diverse areas like: cognitive maps information processing in the brain [3], artificial collective behavior [4], neural networks [5], cellular automata [6], and many others.

Distributed computation exists in all these different systems, whether the system was explicitly designed to compute or not or whether it appears to be performing any useful function. This is referred to as intrinsic computation [7].

Distributed computation can be defined and observed in terms of memory, communication, and processing:

- Memory refers to the storage of information by an agent or process.
- Communication refers to the transfer of information between one agent or process and another.
- Processing refers to the combination or modification of stored and/or transmitted information into a new form.

Systems performing any kind of distributed computation can be observed and implemented as a multi-agent system. Multi-agent system's characteristics like lack of central control, agent behavior and communication possibilities make multi-agent systems very applicable to the distributed computation systems. In this paper a formal model for distributed computation based on multi-agent system is used to build one such system to investigate whether that model is appropriate.

3 Multi-Agent Systems

Multi-agent system (MAS) is a network of entities (called agents) working together on solving the problem that is beyond the agent's individual solving capabilities and knowledge [8]. An agent is software or hardware system situated in some kind of an environment from which it can receive stimulus and flexibly and autonomously act in pursuing its goals [8, 9]. Agent environment can be anything and in the case of distributed computation environment is limited practically just to other agents involved in distributed computation. Each agent has a partial information or problem solving capability while the global system control does not exist in MAS. MAS is simply a group of agents, with afore-mentioned properties, that communicate [10].

Distributed computation observed in terms of memory, communication, and processing in a multi-agent system is defined with agent behavior and communication with other agents.

Communication is the foundation of the multi-agent system. Agents need to share common agent communication language (ACL) and common ontology to be able to communicate [11]. Majority of existing multi-agent systems today rely on the Foundation for Intelligent Physical Agents (FIPA) standards. FIPA is IEEE Computer Society standards organization dedicated to the promotion of technologies and interoperability specifications that facilitate the end-to-end interworking of intelligent agent systems in modern commercial and industrial settings. To create, design or realize an agent, agent behavior has to be identified.

4 Multi-Agent Systems in Distributed Computation

Generic model for a multi-agent system sustaining different distributed computation systems is proposed. It is called a distributed computation multi-agent system (DCMAS). In short a DCMAS is a multi-agent system with agents performing together a distributed computation. As already stated distributed computation is a broad term. Different distributed computation systems exist, but suggested model is generic enough to be applicable to any kind of distributed computation. In the next section proposed model is applied to a Fuzzy Cognitive Map distributed computation system.

Model is defined setting up model of agents that build up a distributed computation multi-agent system. Agents are defined with internal and external agent behavior. Internal behavior is defined as behavior an agent performs to do a partial computation that agent is responsible for. Agent internal behavior covers memory and processing in distributed computation system.

Overall distributed computation task has to be assembled from partial computations so agents need coordination and interaction defined with agent's external behavior. External behavior implements coordination among agents required to perform overall computation task that is distributed among agents. Agent external behavior covers communication in distributed computation system.

Each agent in a distributed computation multi-agent system is defined as a finite state machine. Agent internal and external behavior is implemented with agent states and state transition events.

Definition 1

The DCMAS agent is mathematical defined as a 7-tuple $ADMAS=(S, U, I, P, S_{in}, \omega, \delta)$ where:

S is the finite non-empty set of agent's states, the union of sets S_{local} and $S_{external}$, $S = S_{local} \cup S_{external}$, U is the finite input alphabet for $ADMAS$, a set of agent ACL messages and defined agent internal symbols, I is the finite output alphabet for $ADMAS$, a set of agent ACL messages and defined agent internal symbols, P is the finite set of agent's internal behavior conditions, S_{in} is the set of agent initial states, $S_{in} \subset S$, ω is the agent output function, $\omega: S \times (U \cup P) \rightarrow I$,$\delta$ is the agent next-state transition function, $\delta: S \times (U \cup P) \rightarrow S$.

Agent's states that don't generate events affecting other agents are called local states, denoted with S_{local}. The agent participating in the distributed computation is affecting other agents participating in the same distributed computation by sending ACL messages. Local states are agent states in which an agent does not generate outgoing ACL massages and does not influence other agents. External states, denoted with the $S_{external}$, generate events that affect other agents participating in the same distributed computation. External states are agent states in which an agent generates outgoing ACL massages and influences other agents. Local state can generate state transition event to an external state. The local-external state influence goes both ways, i.e. external state can trigger local state transition. That is way S_{local} and $S_{external}$ are joined in one finite non-empty set of agent states S. The state dependencies between local and external states in one agent and between external states among agents are dynamically specified at run time from a set of predefined dependency types. Each agent stores in its local knowledge database definition of the transitions events for the local states, definition of the transitions events for the external states and definition of the transition events between local and external states. An agent can transit from the local to the external state and vice versa. Agent's states and transition events define distributed computation multi-agent system agents communication, coordination and cooperation, as well as agent internal behavior. Agent internal symbols are used in the state transitions that are not caused by other agents. These symbols are stored in the local knowledge base in each of the agents, together with the ACL messages the agent can receive or send. Agent's internal behavior conditions, denoted with the *P*, are conditions that can occur during the execution of an agent behavior and can cause generation of the output symbol to other agents and/or state transition event in the observed agent. Internal behavior conditions are not caused by the state transition events from the observed agent or from other agents participating in the distributed computation. These conditions are defined within agent behavior, in some of agent states, without influence from other states or input symbols.

A DCMAS agent design requires the definition of local and external states, agent internal symbols, agent's behavior conditions and ACL messages exchanged between the agents participating in the same distributed computation. That knowledge is stored in the local knowledge base in each of the agents. The external state is defined as a state in which agent can create and send ACL messages to the other agents and can receive ACL messages. The local state is a state in which agent can not send ACL messages to the other agents, but can receive ACL messages. Agent internal symbols are local for one agent and do not affect other agents. Also the agent's internal behavior conditions are local for one agent. The behavior conditions are not caused by any state transition but they cause state transition and generate the output alphabet symbol.

5 Case Study

Case study is based on inclusion of multi-agent system in a Fuzzy Cognitive Map method of distributed computation. FCM is a qualitative modeling method and system dynamic behavior technique. Each map represents a system as a group of concepts and cause-effect relations among those concepts [12] depicted with a weighted, directed graph. All values (of concepts and cause-effects) in a map must be mapped to the [-1, 1] interval. The positive weighted factor sign indicates that if the concept from which cause-effect relation originates increases, then the concept in which cause-effect relation terminates also increases while the negative weighted factor sign indicates that if the concept from which cause-effect relation originates increases, then the concept in which cause-effect relation terminates decreases. The larger the cause-effect relation weighted factor, the stronger the cause-effect relation among concepts. FCM inference process is a manipulation of two matrices [12, 13], the concept vector, containing nodes values, and the adjacency matrix, containing cause-effects values among nodes. The value of *i-th* node at the discrete time step *t* is marked A_i^t, the *i-th* column element in the adjacency matrix is marked w_{ji}, then A_i^t is calculated according to the

$$A_i^t = f\left(\sum_{j=1,j\neq i}^{n} A_j^{t-1} w_{ji}\right) \quad (1)$$

FCM is a discrete distributed computation system. Each node carries information about its own state. Information on the overall system state is distributed among nodes. The computation is carried out in discrete time steps.

The ABFCM (Agent Based Fuzzy Cognitive Map) is a distributed computation multi-agent systems obtained with the one-way mapping function from the FCM concepts set to the agent set [14]. Each concept is translated into exactly one agent. Agents in the ABFCM have subsumption architecture so the agent behavior is broken down in to a hierarchy of simple behaviors (Fig. 1.). Behaviors are derived from FCM properties and multi-agent systems properties to provide an multi-agent system that performs distributed calculation like a FCM.

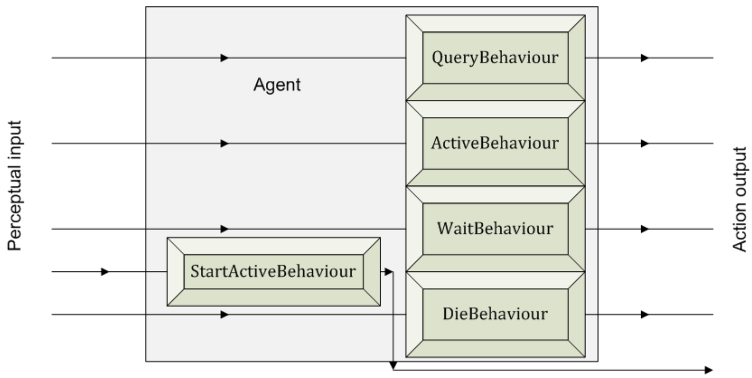


Fig. 1. ABFCM agent's hierarchical architecture

Prototype of the ABFCM is realized on the JADE agent framework [15]. As defined in the previous chapter each agent in a computation multi-agent system is observed as a state machine with the states and transition events. Agent behaviors (Fig. 1.) are realized as machine states so for this computation multi-agent system each agent has five states. Agent states and possible transitions are shown in the next Figure (Fig. 2.).

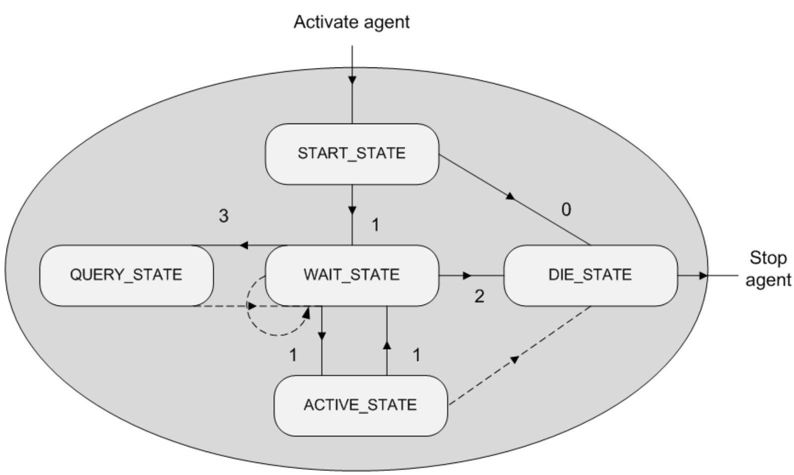


Fig. 2. ABFCM agent's states and transitions

Dashed line transitions are default transitions. Agent states and transitions define ABFCM distributed computation multi-agent system agent.

The ABFCM distributed computation multi-agent system agent according to the Definition 1. is a 7-tuple ABFCMDMAS=(S, U, I, P, Sin, ω, δ) where:

$S = S_{local} \cup S_{external}$, S_{local} ={WAIT_STATE}, $S_{external}$ ={START_STATE, QUERY_STATE , ACTIVE_STATE, DIE_STATE}

U ={Answer, Query, Change, Die, 0, 1, 2, 3}
I={Answer, Query, Change, Die, 0, 1, 2, 3}
P={error free behavior completion, error, message generation}
S_{in} = {START_STATE}
ω is the agent output function, $\omega: S \times (U \cup P) \rightarrow I$
ω(START_STATE, error)= 0
ω(START_STATE, message generation)= Change
ω(START_STATE, error free behavior completion)= 1
ω(WAIT_STATE, Change)= 1
ω(WAIT_STATE, Die)= 2
ω(WAIT_STATE, error)= 2
ω(WAIT_STATE, Query)= 3
ω(ACTIVE_STATE, error)= 0
ω(ACTIVE_STATE, error free behavior completion)= 1
ω(ACTIVE_STATE, message generation)= Query
ω(QUERY_STATE, error free behavior completion)= 0
ω(QUERY_STATE, message generation)= Answer
ω(DIE_STATE, message generation)= Die
δ is the agent next-state transition function, $\delta: S \times (U \cup P) \rightarrow S$
δ(START_STATE, error)= DIE_STATE
δ(START_STATE, error free behavior completion)= WAIT_STATE
δ(WAIT_STATE, Change)= ACTIVE_STATE
δ(WAIT_STATE, Die)= DIE_STATE
δ(WAIT_STATE, error)= DIE_STATE
δ(WAIT_STATE, Query)= QUERY_STATE
δ(ACTIVE_STATE, error)= DIE_STATE
δ(ACTIVE_STATE, error free behavior completion)= WAIT_STATE
δ(QUERY_STATE, error free behavior completion)= WAIT_STATE

Answer, Query, Change and Die are ACL performatives that agents use to inform each other. 0, 1, 2 and 3 are transitions' symbols. Agent's internal behavior conditions are error free behavior completion, error and message generation.

If realized ABFCMDMAS has adequate properties, it will behave like classic FCM. To test this we took existing map [16] (Fig. 3.), converted it to ABFCMDMAS and generate results. The map is used as medical diagnostic decision support system. Decision support system (DSS) is an information system that supports decision making [17]. This map is used to help medical professionals to give a medical diagnose. The map

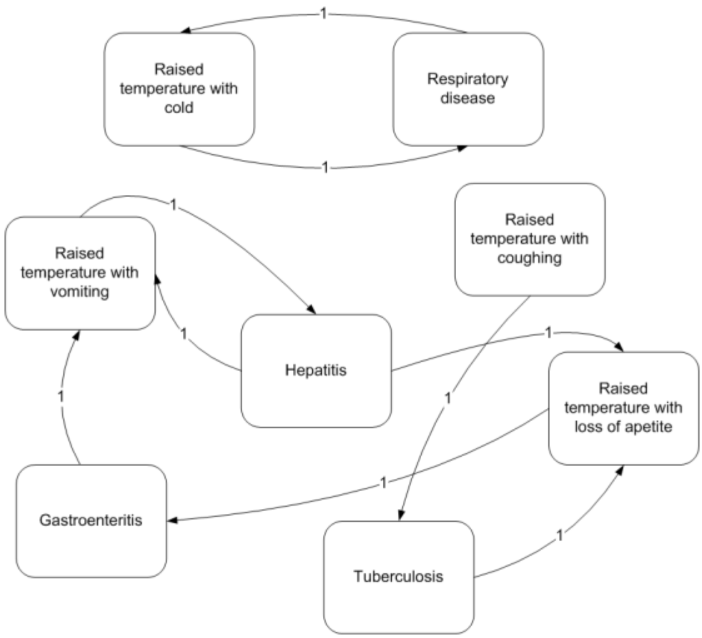


Fig. 3. FCM for decision support in medical diagnosis

concepts are illness symptoms and illness causing those symptoms. If given symptoms as entered in the map as an input vector, the map concludes which illness is in question.

The adjacency matrix for the map is:

$$W = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2}$$

If initial vector is $A^{t=0}$=[1 0 0 0 0 0 0 0], we are looking for the map conclusion on illness with symptom raised temperature with cold. Results are presented in the Table 1. ABFCMDMAS generates the same conclusion as a classic FCM i.e. map conclusion is respiratory disease.

If initial vector is $A^{t=0}$=[0 0 0 1 0 0 0 0], we are looking for the map conclusion on illness with symptom raised temperature with loss of appetite. Results are presented in the Table 2. ABFCMDMAS generates the same conclusion as a classic FCM i.e. map conclusion is gastroenteritis or hepatitis or tuberculosis.

Table 1. ABFCMDMAS results for initial vector is $A^{t=0}$=[1 0 0 0 0 0 0 0], A_1 is raised temperature with cold, A_2 is raised temperature with vomiting, A_3 is raised temperature with loss of appetite, A_4 is raised temperature with coughing, A_5 is respiratory disease, A_6 is gastroenteritis, A_7 is hepatitis, A_8 is tuberculosis

Time step t	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
0	1	0	0	0	0	0	0	0
1	1	0	0	0	1	0	0	0
2	1	0	0	0	1	0	0	0

Table 1. ABFCMDMAS results for initial vector is $A^{t=0}$=[0 0 0 1 0 0 0 0]

Time step t	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
0	0	0	0	1	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	1	0	0	0	1
3	0	0	1	1	0	1	0	1
4	0	1	1	1	0	1	0	1
5	0	1	1	1	0	1	1	1
6	0	1	1	1	0	1	1	1

For all initial vectors the map is shown to generate identical conclusion as a classic FCM.

6 Conclusion

Distributed computation systems can be built upon a multi-agent system. In fact multi-agent systems are quite suitable for implementation of any distributed computation system due to the multi-agent systems' intrinsic properties, primarily agent autonomy and communication.

Proposed generic model for distributed computation multi-agent system (DCMAS) is defined through finite state machine with agent's internal and external behavior. The model is tested through development of a Fuzzy Cognitive Map distributed computation. Obtained results have shown that built ABFCMDMAS behaves exactly like the classic FCM. Distributed computation multi-agent system model upon which ABFCMDMAS was built has shown expected results in FCM distributed computation.

Further work is necessary to confirm model correctness with variety of distributed computation systems. It will be directed in implementing proposed multi-agent model in other distributed computation systems. Also further work will be guided into investigating additional characteristics of distributed computation systems that could be obtained with multi-agent systems.

References

1. Zhong, Z., McCalley, J.D., Vishwanathan, V., Honavar, V.: Multiagent system solutions for distributed computing, communications, and data integration needs in the power industry. IEEE Power Engineering Society General Meeting 1, 45–49 (2004)
2. Stula, M., Stipanicev, D.: Monitoring and control of complex, distributed systems using agent technology. In: Proceedings of International Conference on Software in Telecommunications and Computer Networks, Croatia, Italy, pp. 347–354 (1999)
3. Axelrod, R.: Structure of Decision. In: The Cognitive Maps of Political Elites. Princeton University Press, USA (1976)
4. Demazeau, Y.: From Interactions to Collective Behaviour in Agent-Based Systems. In: Proceedings of the 1st European Conference on Cognitive Science, pp. 117–132 (1995)
5. Ripley, B.D.: Pattern Recognition and Neural Networks, Cambridge (1996)
6. Ilachinsky, A.: Cellular Automata. World Scientific Publishing (2001)
7. Feldman, D.P., McTague, C.S., Crutchfield, J.P.: The organization of intrinsic computation: Complexity-entropy diagrams and the diversity of natural information processing. Chaos 18(4), 59–73 (2008)
8. Wooldridge, M., Jennings, N.: Intelligent Agents: Theory and Practice. Knowledge Engineering Review 10(2), 115–152 (1995)
9. Woodridge, M.: An Introduction to MultiAgent Systems. John Wiley & Sons, New York (2002)
10. Ferber, J.: Multi-agent Systems, An Introduction to Distributed Artificial Intelligence. Addison-Wesley, Reading (1999)
11. Gruber, T.R.: Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal of Human Computer Studies 43(5-6), 907–928 (1995)
12. Kosko, B.: Fuzzy cognitive maps. Int. J. Man-Machine Studies 24, 65–75 (1986)
13. Stylios, C.D., Groumpos, P.P.: Fuzzy Cognitive Map in Modeling Supervisory Control Systems. J. of Intelligent & Fuzzy Systems 8(2), 83–98 (2000)
14. Štula, M., Stipaničev, D., Bodrožić, L.: Intelligent Modeling with Agent-Based Fuzzy Cognitive Map. International Journal of Intelligent Systems (25), 981–1004 (2010)
15. Bellifemine, F., Caire, G., Greenwod, D.: Developing Multi-Agent Systems with JADE. John Wiley & Sons, New York (2007)
16. Kandasamy, W.B.V., Smarandache, F.: Fuzzy Cognitive Maps and Neutrosophic Cognitive Maps. Books on Demand, ProQuest Information & Learning, USA (2003)
17. Power, D.J.: What is a DSS? DSstar, The On-Line Executive Journal for Data-Intensive Decision Support 1(3) (1997)

Effective Graph Representation for Agent-Based Distributed Computing

Adam Sędziwy

AGH University of Science and Technology, Department of Automatics,
al. Mickiewicza 30, 30-059 Kraków, Poland
sedziwy@agh.edu.pl

Abstract. A model decomposition allows transforming a graph-based problem into a set of subproblems to be processed in parallel. Prepared model should guarantee good efficiency of communication and synchronization among agents managing a distributed representation of a system. In this article we present the new approach to such a decomposition, reducing required cooperation among agents. The comparison with replicated complementary graphs approach is also presented.

Keywords: graph, slashed form, distributed computing, multiagent system, lighting control.

1 Introduction

Graph structures provide a flexible and widely used modeling framework for solving various types of problems in such areas as system specification, software generation or task allocation control [5,11]. Moreover, graph transformations may be used to model dynamics of systems. The limitation for their applicability is the time complexity of the parsing or membership problems. That difficulty may be overcome however by using distributed computations paradigm or by decreasing an expressive power of a graph grammar if possible. GRADIS (GRAph DIStributed) multiagent environment capable of performing such distributed graph transformations (representing both algorithmic and algebraic approaches) was presented in [6,9].

Multiagent systems are the effective approach to solving computer aided design (CAD) related problems in such areas as the automotive industry [2], a support in constructional tasks [13], collaborative CAD systems [10] or the adaptive design [8]. The domain lying on the border of CAD and control problems is large-scale intelligent lighting [12]. Large-scale intelligent lighting (abbrev. LaSIL) problem consists of two, subsequent subproblems. The first one is finding a distribution of lighting points in an urban area. This distribution has to satisfy given criteria that include covering a considered area with a suitable luminosity and, on the other side, minimizing exploitation costs generated mainly by a power consumption. The second subproblem is designing an intelligent, adaptive lighting control, supported by distributed sensors, providing information about an environment state. The similar problem, namely computer simulation of an

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 638–647, 2012.

adaptive illumination facility was solved using environment descriptions based on cellular structures (Situated Cellular Agents and Dissipative Multilayered Automata Network) [1]. The weakness of this approach however is the limited applicability to problems formulated above.

The main difficulty related to LaSIL is its high computational complexity. In the first phase (i.e. while setting up an optimal distribution of lamps) it consists of multiple optimization subproblems in 3D space. In the second phase (lighting control), optimizations following environment changes are performed. Note that the second phase is perpetual.

As the computations mentioned above may be broken into separable (or weakly coupled in overlapping areas) local tasks, the first step to be done is finding an appropriate representation of LaSIL, enabling a problem decomposition. The graph formalism seems to be the most convenient one for solving LaSIL due to its correspondence with the problem structure and the mentioned capability of modeling system architecture and dynamics (e.g. environment changes). After decomposing a graph formulation of a problem into a set of smaller subtasks a multiagent system is deployed on it to solve a problem in parallel.

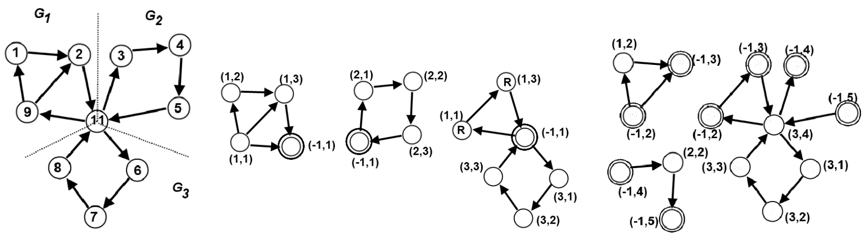
Since the specificity of LaSIL problem eliminates the need of replicating of fragments of subgraphs, we replace the formalism of RCG (Replicated Complementary Graphs) [7] used in the GRADIS environment with the *slashed representation* approach which does not support replication as RCG one does, but reduces and simplifies a cooperation between agents and thereby improves a system performance and dependability.

The article is organized as follows. In Section 2 the concept of replicated complementary graphs is sketched. Section 3 introduces the notion of the *slashed form* of a graph and the related algorithms. Section 4 describes Incorporate procedure for a slashed representation. Section 5 presents the performance comparison for RCG and slashed representations. Section 6 contains the discussion on an effectiveness issues. Final conclusions are presented in Section 7.

2 Related Work - Replicated Complementary Graphs

Replicated Complementary Graphs (RCG) concept enables a decomposition of a centralized graph model of a system into a set of subgraphs (so called *complementary graphs*) with replication of some fragments of those graphs. Decomposition is based on a recursive splitting of a centralized graph. Split procedure may be described as follows. A given graph G is divided into two subgraphs G_1, G_2 in such a way that nodes which are shared by G_1 and G_2 (so called *border nodes*) are replicated together with edges connecting them with other border nodes common for both subgraphs. Borders between RCGs obtained by a decomposition of a centralized graph may change in a result of *incorporating* one or more border nodes by agents maintaining particular complementary graphs. The detailed description of Incorporate procedure for that approach may be found in [9]. In certain circumstances the replication of a given fragment may be performed instead of incorporating it. For example in the case when an agent needs to know a neighborhood of a given vertex and no changes are made on that neighborhood. That results in a lower complexity expressed in a number of exchanged messages.

The exemplary graph and its replicated complementary form are shown in Figure 1. The RCG form of G has only one border node shared by all subgraphs (node indexed with $(-1,1)$ in Fig.1(b), marked with the double circle). Nodes $(1,1),(1,3)$ belonging to G_1 are replicated and attached to G_3 together with the connecting edge and two edges incident to the vertex indexed with $(-1,1)$. They are labeled with R in G_3.



(a) (b) (c)

Fig. 1. (a) G in the centralized form. The dotted line marks planned borders of decomposition (b) $\{G_1, G_2, G_3\}$ - the complementary form of G (c) The complementary form of G after incorporating $(-1,1)$ into G_3

Figure 1c presents the complementary form of G (shown in Fig.1b) after incorporating $(-1,1)$ into G_3. Performing that operation required locking replicas of $(-1,1)$ together with adjacent nodes: $(1,1),(1,3),(2,1),(2,3),(3,1),(3,3)$. After completing the operation the reindexation of corresponding nodes was made.

3 Slashed Form of Centralized Graph

The concept of the slashed form of a centralized graph aims reducing coupling among subgraphs (generated by border nodes) in a distributed representation and thereby simplify operations performed by maintaining agents in a distributed environment. The basic idea of that approach is splitting edges rather than the multiple replication of existing nodes of a centralized graph as it was made in RCG environment. The formal definitions are presented below.

Definition 1. *(Σ^v, Σ^e, A)-graph is a triple $G = (V, E, \varphi)$ where V is nonempty set of nodes, $E \subseteq V \times (\Sigma^e \times A) \times V$ is a set of directed edges, $\varphi : V \longrightarrow \Sigma^v$ is a labeling function, Σ^v and Σ^e denote sets of node and edge labels respectively and A is a set of edge attributes. We denote the family of (Σ^v, Σ^e, A)-graphs as $\mathcal{G}$.*

Note that Definition 1 modifies the (Σ^v, Σ^e)-graph notion (see [7]). We change the edge structure from $V \times \Sigma^e \times V$ to $V \times (\Sigma^e \times A) \times V$ to enable storing all required data in edge attributes. These data include slashing details (e.g. geometric coordinates) but also problem specific information (e.g. architectural details of adjacent buildings). (Σ^v, Σ^e)-graph definition can be also extended e.g. by introducing an attributing function for nodes, but such an extension does not impact further considerations so it will not be considered here.

Definition 2 (Slashed form of G). *Let $G = (V, E, \varphi) \in \mathcal{G}$. A set $\{G_i\}$ of graphs is defined as follows.*

- *$G_i = (V_i, E_i, \varphi_i) \in \mathcal{G}$ and $V_i = C_i \cup D_i, C_i \cap D_i = \emptyset$, where C_i is a set of* ***core nodes****, D_i denotes a set of* ***dummy nodes*** *and $\varphi_i \equiv \varphi|_{V_i}$,*
- *$\bigcup_i C_i = V$ where C_i, C_j are mutually disjoint for $i \neq j$,*
- *$\forall v \in D_i \, \exists ! v' \in D_j (i \neq j)$ such that v' is the replica of v; $\forall v \in D_i \deg(v) = 1$,*
- *$\forall e \in E_i :$ e is incident to at last one dummy node.*

An edge incident to a dummy node is called a ***border edge****. The set of all border edges in G_i is denoted as E_i^b. A set $E_i^c = E_i - E_i^b$ is referred to as a set of* ***core edges*** *of G_i. Let $M = \Sigma^e \times A$, then a set $\{G_i\}$ as defined above is referred to as a* ***slashed form*** *of G, and denoted $\not{G}$, iff following conditions are satisfied:*

1. *$\forall G_i^c = (C_i, E_i^c, \varphi_i|_{C_i})$, $\exists H_i \subset G : H_i \overset{\alpha}{\simeq} G_i^c$ (α denotes an isomorphic mapping between graphs) and H_i, H_j are disjoint for $i \neq j$.*
2. *$\exists f : M^2 \to M$ – a bijective mapping $\forall (e, e') \in E_i^b \times E_j^b \, (i \neq j)$ such that (i) $e = (x_c, m, v) \in C_i \times M \times D_i$, $e' = (v', m', y_c) \in D_j \times M \times C_j$, (ii) v' is a replica of v : $\exists ! \, e_{ij} = (x, m_e, y) \in E$ such that $x_c = \alpha(x), y_c = \alpha(y)$ and $f(m, m') = m_e$. e_{ij} is called a* ***slashed edge*** *associated with replicated dummy nodes v, v'.*
3. *$\forall e = (x, m, y) \in E$: (i) $\exists ! \, e_c \in E_i^c$ for some i, such that $e_c = \alpha(e)$ or (ii) $\exists ! \, (v, v') \in D_i \times D_j$ for some i, j, such that e is a slashed edge associated with v and v'.*

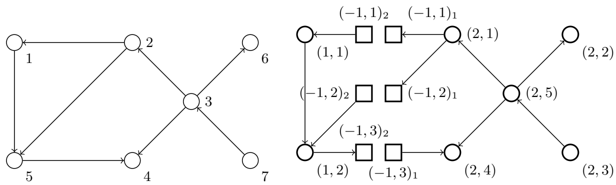
$G_i \in \not{G}$ is called a ***slashed component*** *of G.*

Note that f mapping recovers labeling/attributing data of a slashed edge basing on a labeling/attributing of given border edges.

Remarks. (i) To preserve the clarity of images we neglect the attributing/labeling of graph edges in figures. (ii) A border edge incident with a dummy vertex v will be denoted as e_v. Similarly, a border edge incident with a dummy node indexed with an index id will be denoted as e_{id}.

In Figure 2 the centralized and the slashed form of the given graph G are shown. Core nodes are marked as circles and dummy ones as squares. The following indexing convention is used for slashed components (see Fig. 2b). A core node index has the form (i, k) where i is an unique, within $\not{G}$, identifier of a slashed component G_i and k is an unique, within G_i, index of this node. A dummy node index has the form $(-1, k)_r$ where k is a globally unique identifier of a node. Additionally, a subscript r denotes a reference to a slashed component (or its maintaining agent) hosting a replica of a given dummy node. Using such a subscript allows for immediate localization of a replica. To simplify the notation subscripts will be neglected within the text, unless needed. Note that a dummy vertex and its replica share a common index and differ in reference subscripts only: $(-1, k)_{r_1}, (-1, k)_{r_2}$.

Switching between Centralized and Slashed Representation. Definition 2 introduces the formal background for decomposition of a graph belonging to $\mathcal{G}$



(a) (b)

Fig. 2. (a) Graph G (b) $\not{G}$ representation

but it's poorly applicable in a practical use. Two algorithms, Split and Merge, underlying switching between centralized and slashed representations are introduced below.

Algorithm 1. Split(G, V_c)

---

input : $G = (C \cup D, E, \varphi)$ – graph to be decomposed, $V_c \subset C$ – a subset of the set of core nodes

output: $\not{G} = \{G_1, G_2\}$ where $G_i = (C_i \cup D_i, E_i, \varphi_i)$

```
begin
    C1 ← Vc; D1 ← all dummy nodes from D, adjacent to Vc in G;
    E1 ← all edges connecting nodes from C1 ∪ D1 in G;
    φ1 ← φ|C1∪D1;
    C2 ← V − Vc; D2 ← all dummy nodes from D, adjacent to C − Vc in G;
    E2 ← all edges connecting nodes from C2 ∪ D2 in G;
    φ2 ← φ|C2∪D2;
    Econn ← all edges from E connecting Vc and C − Vc in G;
    foreach Edge e = (x, me, y) ∈ Econn do
        Create dummy node v and its replica v'; Set φ1(v), φ2(v');
        (m, m') ← f^−1(me); ev ← (x, m, v), ev' ← (v', m', y) [1];
        D1 ← D1 ∪ {v}; E1 ← E1 ∪ {ev}; D2 ← D2 ∪ {v'}; E2 ← E2 ∪ {ev'};
return {G1, G2}
```

Split. It is assumed that $G = (C \cup D, E, \varphi) \in \mathcal{G}$ is given, where C is a set of core nodes, D denotes a set of dummy ones and $V_c \subset C$. Initially, for G representing a main graph, $D = \emptyset$. Algorithm 1 presents splitting $G \in \mathcal{G}$ into two slashed components G_1, G_2 (i.e. $\not{G} = \{G_1, G_2\}$) according to the given set V_c of core nodes. To obtain a deeper decomposition, the Split procedure has to be applied recursively on G. The time complexity of Algorithm 1 is $\mathcal{O}(|E|) = \mathcal{O}(|C \cup D|^2)$. In a result of performing the Split procedure on G_i a new component is produced. Replicas of dummy vertices which have been moved form G_i to this new component have to be requested to change references of hosting components in their indices.

Merge. Having a slashed form of G, $\not{G} = \{G_1, G_2, \ldots G_n\}$, one can reassemble a centralized representation, G. It may be accomplished by iterative calls of the Merge procedure described by Algorithm 2 which matches replicas of particular dummy nodes and corresponding border edges. The time complexity of Algorithm 2 is $\mathcal{O}(|E|) = \mathcal{O}(|C \cup D|^2)$.

Algorithm 2. Merge(G_i, G_j)

input : G_i, G_j – slashed components $G_i = (C_i \cup D_i, E_i, \varphi_i)$ to be merged
output: G_{ij} – graph obtained as a result of merging input ones

begin
- $D_i^{common} \leftarrow \{v \in D_i : \exists v' \in D_j : v'$ is the replica of $v\}$;
- $D_j^{common} \leftarrow$ replicas of D_i^{common} in G_j;
- $C_{ij} \leftarrow C_i \cup C_j$; $D_{ij} \leftarrow (D_i - D_i^{common}) \cup (D_j - D_j^{common})$;
- $E_{ij} \leftarrow (E_i - \{$Edges of G_i incident to $D_i^{common}\}) \cup (E_j - \{$Edges of G_j incident to $D_j^{common}\})$;
- $\varphi_{ij} \leftarrow (\varphi_i \cup \varphi_j)|_{C_{ij} \cup D_{ij}}$;
- **for** $e_v = (x, m, v) \in E_i, e_{v'} = (v', m', y) \in E_j$ *where* v' *is the replica of* v [1] **do**
  - $m_e \leftarrow f(m, m')$; $E_{ij} \leftarrow E_{ij} \cup \{(x, m_e, y)\}$;

return $G_{ij} = (C_{ij} \cup D_{ij}, E_{ij}, \varphi_{ij})$

4 Incorporate Procedure for Slashed Representation

Incorporate procedure is the most frequently performed operation in an agent system being considered. For this reason its efficiency impacts strongly on the overall system performance. A slashed component's border consisting of all its dummy nodes, may be shifted by calling Incorporate on a given border edge e_v incident with a dummy vertex v. It is accomplished by matching v and its replica v' being an endpoint of a border edge $e_{v'}$ in another component and recovering an underlying edge form e_v and $e_{v'}$. Thus exactly one core node and possibly some dummy ones are attached to a given component. Figure 3 presents the slashed form of some G and the change in $\not{G}$ implied by incorporation of the border edge $e_{(-1,1)}$ by G_1 which will be called an initiating component. Let us consider this example in detail. The operation Incorporate(G_i, e_v) for G_1 and the border edge e_v incident with the dummy node $v = (-1, 1)_2$ consists of following steps:

1. Get the replica of v, namely $v' = (-1, 1)_1$, localized in G_2 (as referenced by the subscript of the v's index).
2. Attach the core node u indexed by $(2, 1)$, neighboring v', to G_1 and reindex u to $(1, 3)$ (for the compliance with the indexation in G_1).
3. Attach to G_1 all border edges incident to u. Let us assume WLOG that a border edge $e_d = (u, m, d)$. Two scenarios are possible. In the first one a dummy vertex d (in the example d indexed with $(-1, 2)$) has a replica d' in an initiating graph. In that case d, d' are removed and a corresponding slashed edge e is recovered (in Fig. 3b $e = ((1, 3), m_e, (1, 2))$). In the second scenario d does not match any dummy vertex in an initiating graph and it is attached together with e_d to an initiating graph. Note that e_d may not be split because an edge belonging to a slashed component may be incident with at last one dummy node.

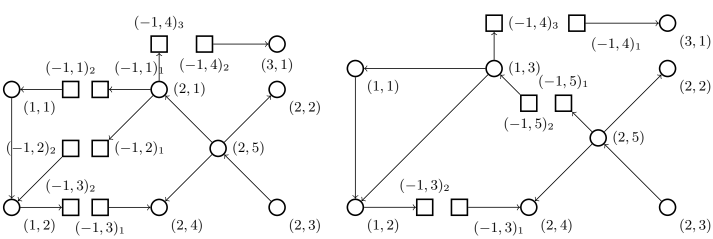
[1] $e_v = (v, m, x)$ and $e_{v'} = (y, m', v')$ if e is directed inversely.

4. Split all edges connecting u with other core nodes (see splitting in Algorithm 1, line 9 and next). Resultant dummy vertices are attached to the initiating graph together with incident edges. The edge $e = ((2,5), m_e, (2,1))$ shown in Figure 3 has been split, resultant $e_{(-1,5)} = ((-1,5), m, (1,3))$ has been attached to G_1 (note that u was reindexed previously) and $e'_{(-1,5)} = ((2,5), m', (-1,5))$ remains in G_2.

Incorporate(G_0, e_v) procedure execution has one phase only. Let us assume that v' is a replica of v and c is a core node being the neighbor of v'. To perform Incorporate(G_0, e_v) an initiator agent (say A_0) sends a `commit` request to a responder agent (say A_1) to gather c vertex and its neighborhood (which may contain newly created dummy nodes). A_1 may either supply requested nodes/edges or reject a request (a response is `locked`) if one or more of requested nodes/edges are already locked by some other initiator. If A_0 was replied with a requested subgraph then it sends `update` requests to agents maintaining replicas of all dummy vertices which just have been attached to G_0. A_0 requests to update references for replicas of those vertices. Note that references to those agents are known to A_0 as they are included in dummy nodes' indices. It is assumed that each an agent keeps the information where his particular dummy nodes were moved to. Hence that if an agent being a recipient of an `update` message has not a dummy vertex replica u' to be updated, because u' node was previously moved to some other slashed component G_k, then it forwards `update` request to the corresponding agent A_k and informs A_0 that a replica location changed. In the case when a response from A_1 is `locked` A_0 repeats the operation with a random delay.

The Incorporate protocol described above is more efficient compared to its form used in the case RCG representations which acts according to the 2PC protocol semantics.

Figure 3b demonstrates an index update: after incorporating $e_{(-1,1)}$ by G_1 the index $(-1,4)_2$ in slashed component G_3 has changed to $(-1,4)_1$.



(a) (b)

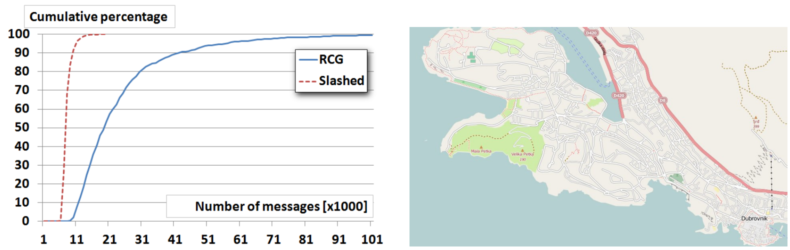
Fig. 3. (a) $\not{G} = \{G_1, G_2, G_3\}$ (b) $\not{G}$ after incorporating $e_{(-1,1)}$ by G_1

5 Tests – Comparing Performances

To compare the effectiveness of RCG and slashed representations we made two tests. In the first one we compared both the number of messages sent by agents during the operation of an optimization of a set of subgraphs and the final number of produced subgraphs. For the better efficiency of a multiagent system populated on this set and performing a LaSIL problem task, a number of included subgraphs should be low. In the second test we compared a convergence of the relaxation algorithm for both approaches.

Number of Messages and Subgraphs. In the first test, the optimization objective was achieving equally sized subgraphs (see [9]). In both cases the following relaxation method was applied. If a given subgraph G_i has an optimal size then no action is performed. Otherwise some neighboring subgraph G_k is selected and incorporated into G_i and next, the resultant G'_i is split into G''_i, G'_k such that $|V(G''_i)| \approx |V(G'_k)|$. Updating messages are sent by a maintaining agent while executing incorporate and split operations. The technical details of the test are following. First, the random 2000-node IE-graph G (see [3]) was generated as an input for the relaxation process performed for both approaches (i.e. for RCG and slashed forms). Next, for each representation, G was decomposed into subgraphs having not more than 3 nodes/core nodes and, in the sequel, the relaxation was performed. The subgraph target size for the system self-optimization process was set to 30 ± 5. The process was interrupted when the ratio of optimized subgraphs reached 90%. Then M_{tot} – a total number of messages sent by the agent system was recorded. Such a single test pass was repeated 1000 times and following statistics was calculated: cumulative frequency distribution for M_{tot}, averaged number of sent messages $M_{tot}^{avg} = \frac{1}{N}\sum_{i=1}^{N} M_{tot,i}$, where $N = 1000$, and the variability range $I_{var} = [n_1, n_2]$ for M_{tot}, where $n_1 = \min_i M_{tot,i}$ and $n_2 = \max_i M_{tot,i}$. Following results were obtained. For slashed representation $I_{var} = [5819, 19165], M_{tot}^{avg} = 7902$, for RCG one $I_{var} = [8571, 204274], M_{tot}^{avg} = 24106$. The cumulative frequency distributions (normalized to 100%) for M_{tot} generated for both RCG and slashed representations are presented in Figure 4a. This chart shows for a given number of messages m, what percent of all test passes, produced a total number of messages less than or equal to m. The test described above shows that even for sparse graph structures like IE-graphs the averaged number of sent messages was reduced by 3.05. The average final numbers of subgraphs, $Q^{avg} = \frac{1}{N}\sum_{i=1}^{N} Q_i$, were equaling $Q_{RCG}^{avg} = 84.4, \sigma_{RCG} = 1.5$ and $Q_{sl}^{avg} = 37.7, \sigma_{sl} = 1.4$ for RCG and slashed representation respectively, where σ is a standard deviation and Q_i is a final number of subgraphs for an ith test pass. Thus the number of agents in the RCG underlaid multiagent system is 2.2 times higher than for the slashed based one.

Convergence. To compare the convergence of the relaxation algorithm for both graph representations, the Dubrovnik city map was used as the case (Fig.4b). Similarly as in the first test, the goal was decomposing centralized graph G_{Dub} corresponding to the selected map into equally sized subgraphs, where the target size was set to 20 ± 4. The order of G_{Dub} was $|V| = 4105$ and the size $|E| = 4547$. Such a decomposition is the preliminary action for LaSIL problems, prior to optimizing or controlling tasks which are performed on resultant subgraphs. The test showed that for the RCG data model the system cannot be optimized. After about 5 iterations the ratio of optimized subgraphs stabilizes around the average $\overline{s} = 55.4\%$



(a) (b)

Fig. 4. (a) Cumulative frequency distribution for total numbers of updating messages sent in optimization process (b) OpenStreetMap of Dubrovnik, Croatia

and fluctuates with the standard deviation $\sigma = 3.4$ (the acceptance threshold for the optimization was set to 90% for both representations). For the slashed form the ratio of optimized subgraphs was reached 91.9% in 310 iterations.

6 Discussion

Let us compare in the formal way the complexities of Incorporate operations in both cases, for RCG and slashed representations. In RCG approach each a border node is shared by at least two and at last k graphs, where k is the total number of complementary graphs (maintaining agents). For this reason an operation of incorporating a single border node implies a cooperation among k agents. Note that such a cooperation requires simultaneous locking border nodes belonging to k (in the worst case) RCGs and their neighbor vertices. Together it makes $\mathcal{O}(d \cdot k)$ nodes in the worst case, where d is a maximal degree of a node belonging to G. If any node cannot be locked then the entire operation fails. A locking considered above is required for granting an exclusive access to those nodes to an incorporating agent. From the statistical point of view that requirement may get difficult to satisfy with an increasing number of agents, k. On the other side retrying a procedure execution generates a message exchange overhead in an agent system.

In the case of a slashed representation of a graph, a completion of Incorporate requires cooperation with one agent only and locking at last p dummy nodes and q core edges where $p + q \leq d$ and d is a maximal degree of a node belonging to G. In the other words it is constant with respect to a number of agents.

For the RCG approach an estimated number of updating messages sent by an incorporating agent to other agents is $\mathcal{O}(d \cdot k^2)$. In the slashed representation case it equals $\mathcal{O}(d)$ i.e. it is constant with respect to a number of agents.

Estimations presented above show that using the slashed representation of G simplifies and improves efficiency of the Incorporate operation by reducing a number of cooperating agents from $\mathcal{O}(k)$ to 2, a number of locked nodes/edges from $\mathcal{O}(d \cdot k)$ to $\mathcal{O}(d)$ and a number of updating messages from $\mathcal{O}(d \cdot k^2)$ to $\mathcal{O}(d)$ compared to RCG model.

7 Conclusions

The efficiency of an agent system is the crucial factor deciding about its applicability in distributed computations, in particular in the LaSIL problem. The approach introduced in the paper allows reducing complexity of distributed operations based on a cooperation among agents, preserving the sufficient expressive power for a problem description. It is achieved by changing the conception of graph borders and thereby reducing a number of exchanged messages. Also the complexity of Incorporate procedure has been reduced significantly. Such the approach does not support the replication but in the case of the LaSIL problem this feature may be neglected.

Acknowledgments. The paper is supported from the resources of Alive & KICing project and NCBiR grant no O ROB 002101/ID 21/2.

References

1. Bandini, S., et al.: Self-organization models for adaptive environments: Envisioning and evaluation of alternative approaches. SIMPAT 18(10), 1483–1492 (2010)
2. Baumgart, S., et al.: PLUG: An Agent Based Prototype Validation of CAD-Constructions. In: The Int'l Conf. on Information and Knowledge Engineering (2006)
3. Flasinski, M.: On the parsing of deterministic graph languages for syntactic pattern recognition. Pattern Recognition 26(1), 1–16 (1993)
4. De Floriani, L., Falcidieno, B.: A hierarchical boundary model for solid object representation. ACM Trans. Graph. 7(1), 42–60 (1988)
5. Kotulski, L.: Supporting Software Agents by the Graph Transformation Systems. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006, Part III. LNCS, vol. 3993, pp. 887–890. Springer, Heidelberg (2006)
6. Kotulski, L.: GRADIS – Multiagent Environment Supporting Distributed Graph Transformations. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 644–653. Springer, Heidelberg (2008)
7. Kotulski, L.: On the control complementary graph replication. In: Models and Methodology of System Dependability. Monographs of System Dependability, vol. 1, pp. 83–95. Oficyna Wydawnicza PW, Wrocław (2010)
8. Kotulski, L., Strug, B.: Distributed Adaptive Design with Hierarchical Autonomous Graph Transformation Systems. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007, Part II. LNCS, vol. 4488, pp. 880–887. Springer, Heidelberg (2007)
9. Kotulski, L., Sędziwy, A.: GRADIS - the multiagent environment supported by graph transformations. SIMPAT 18(10), 1515–1525 (2010)
10. Ligong, X., et al.: Multi-agent Architecture for Collaborative CAD System. In: ICCSIT 2008, pp. 7–11 (2008)
11. Rozenberg, G.: Handbook of Graph Grammars and Computing By Graph Transformation, Foundations, vol. I. World Scientific Publishing Co. (1997)
12. Sędziwy, A., Kotulski, L.: Solving Large-Scale Multipoint Lighting Design Problem Using Multi-agent Environment. In: Key Engineering Materials. Advanced Design and Manufacture IV, vol. 486 (2011)
13. Yabuki, N., Kotani, J., Shitani, T.: A Cooperative Design Environment Using Multi-Agents and Virtual Reality. In: Luo, Y. (ed.) CDVE 2004. LNCS, vol. 3190, pp. 96–103. Springer, Heidelberg (2004)

Multi-agent Based Software Licensing Model for Embedded Systems

Goran Horvat, Damir Šoštarić, and Drago Žagar

Department of Communications
Faculty of Electrical Engineering, J.J. Strossmayer University of Osijek
Kneza Trpimira 2b, 31000 Osijek, Croatia
{goran.horvat,damir.sostaric,drago.zagar}@etfos.hr

Abstract. Recent growth of the embedded system industry, both in size and complexity, resulted in numerous advantages for the consumer equipment and a few problems for the system designers. One of the problems is the software licensing problem in complex embedded systems. The fact that today's embedded systems are evolved into fairly large and complex systems presents the need to effectively manage the embedded software functionality, without the need to recompile and rewrite the firmware components. This problem can be avoided by using fully functional software accompanied by a licensing agent to enforce the license permissions granted to an end user. The licensing agent enforces the permissions stored in encrypted license file located on a Secure Digital memory data card. Another advantage is the fact that the permissions can be updated remotely using a license distribution agent located at the manufacturer's server and uploaded through Ethernet or Internet. Licensing agents applied to various embedded systems have the ability to communicate with other licensing agents and license distribution agent in order to exchange and verify supplied licenses. This paper proposes the software licensing agent model and implements the licensing agent in the existing laboratory based embedded system.

Keywords: embedded system, software licensing, licensing agent, XTEA, encryption, license distribution.

1 Introduction

An embedded system refers to a specialized computer system that focuses on application, software and hardware customization and strict requirement in terms of functionality, reliability, cost effectiveness and volume and power efficiency. According to [1], semiconductor and embedded industry is projected to bloom from $3.25 billion in 2005 to $43.7 billion by 2015. With such attractive growth statistics, the field of embedded systems presents an interesting area of research.

In today consumer equipment embedded systems have evolved into fairly large and multipurpose devices that incorporate versatile functions [2]. This evolution presents another problem to the embedded system designers: how to offer a customized embedded system to the end user without the need for additional firmware customization, which

G. Jezic et al. (Eds.): KES-AMSTA 2012, LNAI 7327, pp. 648–657, 2012.

presents a costly solution for the end user (since the entire firmware has to be rewritten). A more elegant and cost effective solution is the design of fully functionally embedded system, while dynamically managing and delivering functionality according to the user requirements. This feature enables the user to dynamically expand or reduce the functionality by the means of a single license file.

In order to effectively monitor and manage the systems functionality on the fly, this paper proposes the use of a multi-agent architecture. The idea is to integrate licensing agents into embedded systems in order to monitor all license limitation and permissions. The agent observes the embedded system and acts upon the permissions and restrictions stored in a license file. The licensing agent interacts with other licensing agents in the vicinity throughout *Ethernet* commutation in order to verify the integrity of a license file and to acquire new licenses. When a license update is due, the license distribution agent situated on the manufacturer server distributes the licenses to the licensing agents in embedded systems thorough *Ethernet* LAN and/or *Internet* connection. Licensing agents have the ability to interact with each other in order to verify the integrity of the license files and to assure the distribution of the license files to all licensing agents. The license distribution agent interacts with licensing agents in embedded systems to ensure the integrity of the license files. The work in this paper could be described as a novel approach in functionality managing in embedded systems using licensing procedure and multi agent framework support.

To ensure the license integrity from pirating an encryption algorithm was applied to protect the license data. The used encryption algorithm is the eXtended Tiny Encryption Algorithm (XTEA), designed and optimized for microcontroller embedded system use [3]. The means of storing the license file can vary whereas this paper proposes storage using standard *micro Secure Digital* memory card (microSD). The license file stored on memory card can be accessed and updated remotely by a licensing agent using *Ethernet* interface [4].

This paper will demonstrate the implementation of the proposed agent using laboratory based embedded device, alongside with the microSD data card for license file storage. Also the encryption algorithm will be analyzed using cryptanalysis in purpose of determining the strength of the encryption.

In the following section the XTEA algorithm will be depicted and analyzed regarding weaknesses and use as a cipher function for license management. Further on, in Section 3 the licensing model is defined while Section 4 describers the functionality of the proposed agent and the advanced implementation of the proposed agent. Section 5 displays the test results of this system regarding the embedded device implementation while Section 6 gives the conclusion.

2 XTEA Cipher as an Encryption Algorithm

In order to ensure license integrity and to secure the license file upon transferring through network, this paper proposes the use of an encryption algorithm. When choosing an appropriate encryption algorithm for implementation, several aspects has to be taken into consideration. The simplicity of the implementation is the key feature when choosing an appropriate algorithm. The security of the algorithm and its cryptanalysis features are of paramount importance when deciding upon a cipher. One cipher that incorporates all the prerequisites is the Extended Tiny Encryption Algorithm (XTEA) [3].

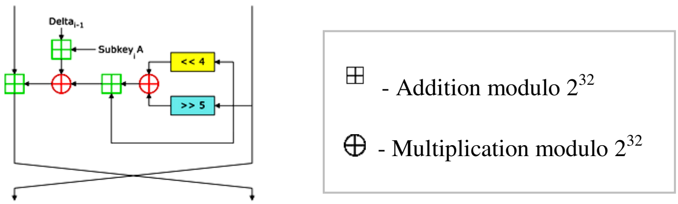


Fig. 1. A *Feistel* rounds of XTEA

The block cipher TEA (Tiny Encryption Algorithm) was designed by Wheeler and Needham in 1994 as a short C language program that would run safely on most machines. It achieves high performance by performing all its operations on 32bit words, using only exclusive-or, addition modulo 2^{32}, multiplication modulo 2^{32} and shifts. TEA has a simple Feistel structure, but uses a large number (i.e. 64) of rounds to achieve the desired level of security [5]. However, taking advantage of its simple key schedule, in 1997 Kelsey, Schneier and Wagner described a related-key attack. To secure TEA against related-key attacks, Needham and Wheeler presented an extended version of TEA in 1997, known as XTEA, which retains the original objectives of simplicity and efficiency [6].

Figure 1 shows one *Fiestel* round of XTEA cipher. The simplicity of the algorithm is seen from the low complexity of the *Fiestel* network structure [3]. In order to implement the cipher into embedded system, *Fiestel* encoder and decoder must be designed using ANSI C programming language. Flowchart of the implemented code is shown in Fig. 2.

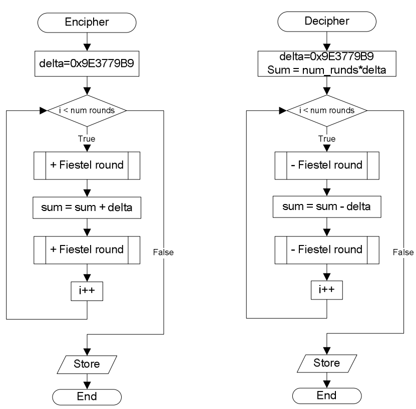


Fig. 2. Flowchart for XTEA Encipher and Decipher functions

The cipher consists of addition, multiplication and shift operators modulo 2^{32}, presenting a simple to implement algorithm in low cost low power embedded systems [3]. Further on, the question that arises is the security of the algorithm. According to

[8], the best cryptanalysis for XTEA is a related-key differential attack. The stated attack can break 32 out of 64 rounds of XTEA, requiring $2^{20.5}$ chosen plaintexts and a time complexity of $2^{115.15}$, which demonstrates the strength of the mathematical algorithm.

However, there are attack techniques that target the physical implementation rather than the algorithm itself [9]. When sampled at high rates and examined in more detail, power waveforms can reveal the key bits [9]. The impact of these attacks is emphasized giving the fact that the attacker is in possession of the device. This presents a security problem in implementing the proposed algorithm. This problem can be overcome using several methods which bring the implementation of this algorithm back to focus. One method is based on using specifically designed and secures integrated circuits (micro-controllers) [10]. The problem using this method is inability to implement the encryption in the existing embedded systems. This can be avoided using second method, according to [11]. The method uses smart random code injection to mask power analysis based side channel attacks. Finally, the problem of reading device's firmware forcefully (to acquire the encryption key) can be avoided using read and write protection applied to the application part of the memory.

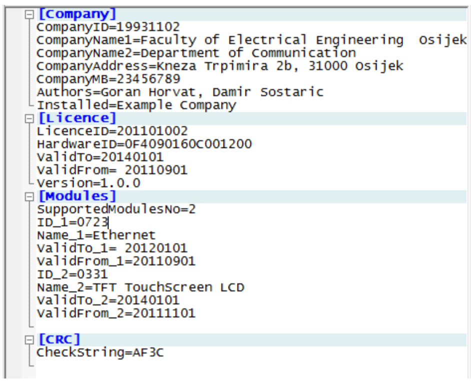
3 Licensing Model for Embedded System

In order to effectively manage the functionality of the embedded system, a license based approach must be taken. Software licensing is any procedure that lets an enterprise or user purchase, install, and use software on a machine or network in accordance with a software vendor's licensing agreement [12, 13]. All software licensing aims to protect both the vendor's investment by minimizing the risk of hard piracy and the enterprise's investment by minimizing the risk of auditing fines from soft piracy. This term used in computer applications can be easily applied to embedded system licensing, due to the fact that the embedded system's firmware is a software component. Having this in mind, standard licensing models can be applied to suit the needs for the embedded system.

On the other hand, the field of embedded software licensing is generally undefined. Related work referring to this topic is poorly represented in the literature and does not present any specific solutions for managing embedded system functionality using a licensing procedure. A review of the software licensing in embedded system is presented in [12] where certain fundamental aspects of embedded software licensing are presented. The main problem presented is the convergence of the embedded system and personal computers where the embedded systems are becoming more like PC and vice versa. On the other hand, the author states that the "Consumers will be hurt by licensing embedded software", which in turn presents a solid claim. However, with the right method of implementation of the licensing method, and the right security conisation the consumers can only benefit from software licensing. In the end, by encompassing both software and hardware components in the licensing management, the possibility of pirating and copying the devices could be reduced, alongside with numerous advantages towards the consumer (end user).

According to [12], variety of licensing models exists than could be applied to the embedded system licensing, depending on the designers policy. In this paper a combination of *Packaged* (Single license purchased for a single user or machine), *Subscription based* (License purchased for some time period) and *Utility-based* (Customer charged according to time product is used) is proposed.

This paper proposes a licensing method based on an encrypted license file stored on standard *microSD* data card, accessed by a licensing agent. The license file is formatted in standard configuration file (*.ini) before the encryption takes place.


```
[Company]
CompanyID=19931102
CompanyName1=Faculty of Electrical Engineering  Osijek
CompanyName2=Department of Communication
CompanyAddress=Kneza Trpimira 2b, 31000 Osijek
CompanyMB=23456789
Authors=Goran Horvat, Damir Sostaric
Installed=Example Company
[Licence]
LicenceID=201101002
HardwareID=0F4090160C001200
ValidTo=20140101
ValidFrom= 20110901
Version=1.0.0
[Modules]
SupportedModulesNo=2
ID_1=0723
Name_1=Ethernet
ValidTo_1= 20120101
ValidFrom_1=20110901
ID_2=0331
Name_2=TFT TouchScreen LCD
ValidTo_2=20140101
ValidFrom_2=20111101

[CRC]
CheckString=AF3C
```

Fig. 3. Unencrypted license file form

As seen in Fig. 3 the unencrypted license file can be opened using a standard text editor and is encoded in standart text format. The file is composed of several sections, where the first section named *Company* incorporates basic company data and only serves the function of vividly representing the company where the license is installed. The second section *Licence* contains basic data for license management such as *LicenceID*, Software Version. Also, this section contains important data such as *HardwareID*, on behalf of which the license is linked to a specific hardware. This *HardwareID* is composed of production lot number, wafer number and wafer coordinates of the specific silicone chip. This uniquely identifies the chip thus eliminating the possibility of license cloning. *HardwareID* field can be extended to a form that a single license file supports several embedded systems (hardware ID's), thus simplifying the licensing on large systems.

Further on, the defined *ValidTo* and *ValidFrom* field represents the Subscription-based licensing model where the license is purchased for some time period [12]. Accordingly, when the license expires the embedded device loses its functionality completely or to a certain degree (depending on the programming).

The section named *Modules* defines additional device functionality for an embedded system, whereas the last section implements the license integrity check in a form of a check string; 16bit CRC string. The primary reason for using check string in unencrypted license file is to circumvent the problems of shuffling or tampering with a portion of the encrypted license file.

After the license file is composed in its unencrypted form, the XTEA encryption is performed and the data is secured. The used encryption key is arbitrary and it is suggested to alter the key depending on the production lot or other factors, minimizing the possibility of piracy. The key scheduling is performed by the manufacturer and the license distribution agent. Generated keys alongside with *HardwareID's* and user credentials are stored in manufacturer's data base and inside embedded system firmware. When a distribution agent needs to deliver a new license to an embedded system (or a group of embedded systems), it will encrypt the license using stored key located in database. Upon delivery, the licensing agent in embedded system will decrypt the license using encryption key stored in the non-volatile memory (protected from reading).

It is important to state that for the *Subscription*-based licensing model the main prerequisite is the existence of the *Real Time Clock* (RTC) module, to ensure time reference.

4 Using Multi-Agent Architecture in Software Licensing

This paper proposes the use of a multi-agent system in order to perform software licensing on remote embedded systems using Licensing agents (Fig. 4). The licensing agents are situated in embedded system and are responsible for enforcing the license permissions and limitations. The licensing agents have the ability to communicate with the other licensing agents in the local network. On the other hand, on the license server a License distribution agent is located whose main purpose is to distribute and verify the license integrity on to licensing agents located in embedded systems. The agents communicate through *Ethernet* interface or through *Internet*.

4.1 Licensing Agent – Embedded System

The licensing agent integrated into the embedded system encompasses the ability to: decode the encrypted data stored as a license file, understand the stated permissions and limitations proposed by the license and to enforce the desired permissions and limitations. Also, the agent needs to have the ability to communicate with other agents in vicinity, and to communicate with License distribution agent. In a case where a direct link to the License distribution agent is not possible, the agent will establish connection through other licensing agents, in order to relay the message.

The proposed agent observes the running software with the accompanied modules. If the flow of the software is diverted to a software module banned from use by the license file (e.g. additional printer module not supported by license) the agent will bypass the unsupported module and resume normal work. If the agent detects error in time reference (tampering), it will contact other agents and correct the malicious attempt. Consequently, if a new license is being distributed from the manufacturer, the license distribution agent (located in the license server, Fig. 4) will contact Licensing agent and deliver a new updated encrypted license file (directly or via other agent).

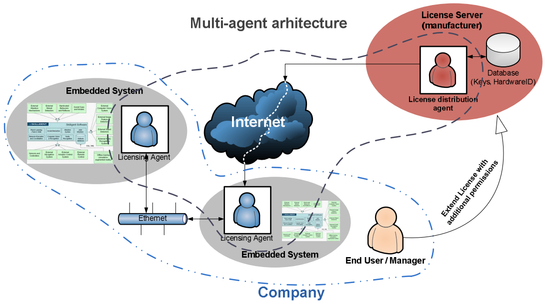


Fig. 4. Multi-agent system – software licensing model

After receiving the license file, the agent performs the license check in the following manner: First, the encrypted license decrypted using incorporated XTEA decryption algorithm with provided key (Fig. 1). After the decrypted license is presented in its original form (Fig. 3) the agent checks the integrity via CRC. Next on, *HardwareID* number and subscription-based fields are verified against device *HardwareID* and RTC. Finally, the additional module support is parsed and examined. After the license if verified with other licensing agents (e.g. by hashing unencrypted license file), the file is stored on *microSD* data card.

This completes the licensing procedure upon receiving new license or upon system startup. If there was an error during licensing procedure (e.g. wrong *HardwareID* or expired time period) the agent will halt the device and the device will not be functional. The interaction protocol of the multi-agent system is shown in Fig.5. The LA defines the licensing agents situated in various embedded systems whereas the License distribution agent is located in the main licensing server.

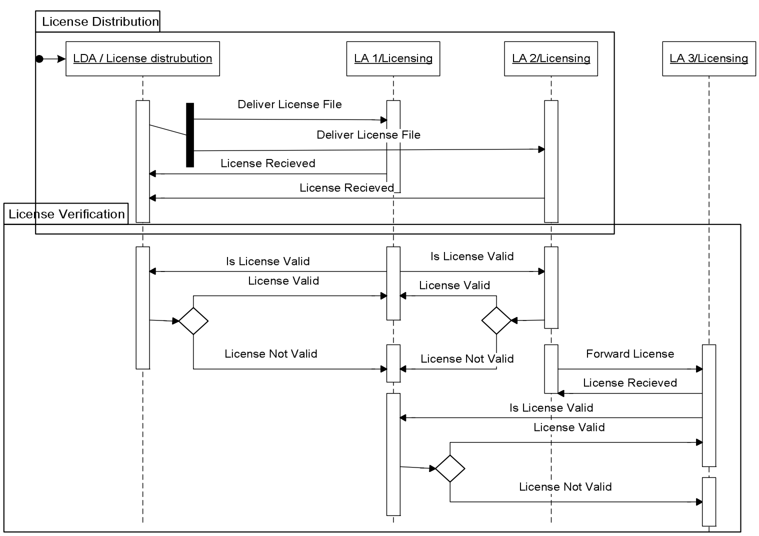


Fig. 5. AUML Diagram of the Multi-agent System

4.2 License Distribution Agent – License Server

In order to effectively distribute the licenses to the end users (e.g. companies, factories etc.) the proposed solution is the license distribution agent. The agent is located on the embedded system manufacture's server (Licensing server) and its primary purpose is delivering and verifying licenses on remote embedded systems.

The main prerequisite to enable the agent to distribute the license files is the knowledge of all existing embedded systems and licensing agents implemented. This primarily means that all existing agents must be in contact with the license distribution agent through *Internet* connection (e.g. Cloud based communication). Second, the agent must have access to the database where all the embedded system's data are stored, such as: primarily XTEA encryption keys written upon production and Company name, *CompanyID* etc. (Fig.3). Once an agent is in possession of this data, it can dynamically recreate license files and deliver them to the desired embedded system.

To give an example (Fig.4), an end user requests an extension of the currently running licenses (to add additional support and to extend the license validation) to the licensing server (WEB shop based or by contacting manufacturer). Once a request is verified (the funds deposited etc.) the License distribution agent will be notified. The agent will then request the encryption key and other required parameters form the database and create new license file. Further on, it will contact the specific Licensing agent where the update is necessary and relay the encrypted license. Once all the licensing agents have decrypted and verified the license file (against each other and license distribution agent), the license becomes valid.

5 Implementation and Testing of the Licensing Agent

In order to effectively test the proposed licensing agent, an existing laboratory model of an embedded system was adopted. The used embedded system is an access control system that utilizes Atmel's AVR XMEGA micro-controller, RTC module, Ethernet module and touch-screen TFT LCD.

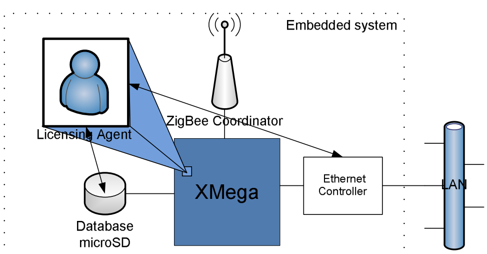


Fig. 6. Laboratory model access control system

However, the problem that arises is how to implement a software agent in low cost low power 8bit micro controller, representing the embedded system. The implementation of an agent into an embedded system requires the existence of *Embedded Operating*

System (EOS) [16]. The used embedded system lacks the main prerequisite for an agent that is the existence of an Embedded Operating System. Accordingly, the implementation of a licensing agent was carried out using interrupt subroutine that embodies the function of a licensing agent. Another thing to note is that this implementation of the agent is incorporated into the existing firmware compiled using *Mikroelektronika's* ANSI C compiler for AVR named MikroC.

Figure 4 displays the used embedded system. Interesting fact to examine of the proposed licensing agent is the overhead that this implementation induces in the existing firmware. To display the induced overhead a statistical analysis of the firmware was performed before and after implemented method.

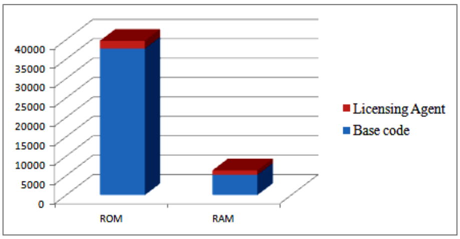


Fig. 7. Induced code overhead for RAM and ROM

The cumulative amount of induced overhead in this implementation is 1994byte for ROM memory and 1122byte for RAM memory. If the induced overhead is inspected regarding the ROM and RAM size of a microcontroller, the induced ROM overhead is 1.5% and induced RAM overhead is 13.8%. The use of precompiled libraries for *microSD* card support will significantly increase the overhead if the existing code does not use these libraries by default.

6 Conclusion

This paper demonstrates an effective way of implementing licensing in embedded systems using multi-agent architecture. The main advantage is flexible managing of the device functionality depending on the future requirements through license support. This paper describes a secure way of encrypting and transferring a license file using easy to implement XTEA cipher. Also, the security issues regarding the cryptanalysis and piracy was addressed and concluded that through various security mechanisms this model can be additionally secured.

The proposed multi-agent system ensures that the licensing process is carried out without the need for intervention in using software agents. With the use of license distribution agent, new licenses can be automatically generated; encrypted using existing key stored in the database and deployed using licensing agents situated on embedded systems. The licensing agents have the ability to communicate with other, thus providing a verification of the transmitted license. With this social component the agents can detect and correct abnormal behavior caused by malicious attackers.

The proposed agent system was implemented partially as a solution for lack of EOS in the tested embedded system. The licensing agent was implemented using standard ANSI C programming language and interrupt subroutines, to mimic the effect of multi threading. Also, the interaction with other software agents was not analyzed and is left for future work and testing. Regarding the induced code overhead to the tested embedded system, it can be stated that the induced overhead of 1.5% is negligible.

Future work involves improving this licensing agent in the direction of using hardware based crypto modules (e.g. integrated hardware AES module). Also, the implementation of the proposed agent in embedded system containing EOS is recommended, alongside with establishing communication with other agents and implementing license distribution agent in license server. The testing of the overall system is suggested alongside with security and performance analysis.

References

1. Parab, J., Shinde, S.A., Shelake, V.G., Kamat, R.K., Naik, G.M.: Practical Aspects of Embedded System Design using Microcontrollers, vol. XXII, p. 150. Springer (2008)
2. Karsai, G., Massacci, F., Osterweil, L.J., Schieferdecker, I.: Evolving Embedded Systems. Computer 43(5), 34–40 (2010)
3. Pavlin, M.: Encription using low cost microcontrollers. In: Proc. of MIDEM - Society for Microelectronics, Electronic Components and Materials, Cop, pp. 189–194 (2006)
4. Horvat, G., Žagar, D., Šoštarić, D.: Ethernet Data Accessing Method for a Micro-controller Embedded Systems. In: Kando Conference, Science in Practice Budapest (2011)
5. Wheeler, D.J., Needham, R.M.: TEA, a Tiny Encryption Algorithm. The Computer Laboratory. Cambridge University (1994)
6. Lu, J.: Cryptanalysis of Block Ciphers. Technical Report RHUL–MA–2008–19, Department of Mathematics Royal Holloway, University of London (2008)
7. Electronic Equipment Manufacturer. MikroElektronika, http://www.mikroe.com
8. Lu, J.: Related-key rectangle attack on 36 rounds of the XTEA block cipher. International Journal of Information Security 8(1), 1–11 (2009)
9. Knežević, M., Rožić, V., Verbauwhede, I.: Design Methods for Embedded Security. TELFOR Journal 1(2) (2009) ISSN 1821-3251
10. Atmel Corporation, http://www.atmel.com (accessed: January 2012)
11. Ambrose, J.A., Ragel, R.G., Parameswaran, S.: A smart random code injection to mask power analysis based side channel attacks. In: Proc. of 5th IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis, pp. 51–56 (2007)
12. Ferrante, D.: Software Licensing Models: What's Out There? IT Professional 8(6), 24–29 (2006) ISBN: 1520-9202
13. Koopman, P.: Embedded Software Licensing? Carnegie Mellon University (October 2000)
14. Kaps, J.-P.: Chai-Tea, Cryptographic Hardware Implementations of xTEA. In: Proc. of the 9th International Conference on Cryptology in India, Kharagpur, India
15. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice Hall, Upper Saddle River (2003) ISBN 0-13-790395-2
16. Ma, X., Qu, Y.: Research on Embedded Agent System Architecture. In: International Conference on Embedded Software and Systems Symposia, ICESS Symposia 2008, July 29-31, pp. 142–145 (2008)
17. Nwana, H.S.: Software Agents: An Overview. Knowledge Engineering Review 11(3), 205–244 (1996)

Author Index