INTELLI 2014

The Third International Conference on Intelligent Systems and Applications

ISBN: 978-1-61208-352-0

June 22 - 26, 2014

Seville, Spain

INTELLI 2014 Editors

Jerzy Grzymala-Busse, University of Kansas, USA

Ingo Schwab, Karlsruhe University of Applied Sciences, Germany

INTELLI 2014

Foreword

The Third International Conference on Intelligent Systems and Applications (INTELLI 2014), held between June 22-26, 2014 - Seville, Spain, was an inaugural event on advances towards fundamental, as well as practical and experimental aspects of intelligent and applications.

The information surrounding us is not only overwhelming but also subject to limitations of systems and applications, including specialized devices. The diversity of systems and the spectrum of situations make it almost impossible for an end-user to handle the complexity of the challenges. Embedding intelligence in systems and applications seems to be a reasonable way to move some complex tasks form user duty. However, this approach requires fundamental changes in designing the systems and applications, in designing their interfaces and requires using specific cognitive and collaborative mechanisms. Intelligence became a key paradigm and its specific use takes various forms according to the technology or the domain a system or an application belongs to.

We take here the opportunity to warmly thank all the members of the INTELLI 2014 Technical Program Committee, as well as the numerous reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to INTELLI 2014. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the INTELLI 2014 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that INTELLI 2014 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of intelligent systems and applications.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the charm of Seville, Spain.

INTELLI 2014 Chairs:

INTELLI Advisory Committee

Michael Negnevitsky, University of Tasmania, Australia
Roy George, Clark Atlanta University, USA
Pradeep Atrey, University of Winnipeg, Canada
Jerzy Grzymala-Busse, University of Kansas, USA
Daniël Telgen, HU University of Applied Sciences Utrecht, The Netherlands
Zoi Christoforou, Ecole des Ponts-ParisTech, France
Jiho Kim, Chung-Ang University, Korea

INTELLI 2014

Committee

INTELLI Advisory Committee

Michael Negnevitsky, University of Tasmania, Australia
Roy George, Clark Atlanta University, USA
Pradeep Atrey, University of Winnipeg, Canada
Jerzy Grzymala-Busse, University of Kansas, USA
Daniël Telgen, HU University of Applied Sciences Utrecht, The Netherlands
Zoi Christoforou, Ecole des Ponts-ParisTech, France
Jiho Kim, Chung-Ang University, Korea
Ingo Schwab, Karlsruhe University of Applied Sciences, Germany
Firas B. Ismail Alnaimi, Universiti Tenaga Nasional, Malaysia
Giuseppe Salvo, Università degli studi di Palermo, Italy
Nittaya Kerdprasop, Suranaree University of Technology, Thailand
Susana Vieira, IDMEC/LAETA, Instituto Superior Técnico, Technical University of Lisbon, Portugal

INTELLI Industry/Research Chairs

Antonio Coronato, National Research Council (CNR) & Institute for High-Performance Computing and Networking (ICAR) - Napoli, Italy
Matjaž Gams, Jožef Stefan Institute - Ljubljana, Slovenia
Antonio Jimeno, NICTA - Melbourne, Australia
Haowei Liu, INTEL Corporation, USA
Michael Affenzeller, HeuristicLab, Austria
Paolo Spagnolo, Italian National Research Council, Italy
Wei Liu, Amazon.com - Seattle, USA
Pieter Mosterman, MathWorks, Inc. - Natick, USA
Paul Barom Jeon, Samsung Electronics, Korea
Kiyoshi Nitta, Yahoo Japan Research, Japan
Wolfgang Beer, Software Competence Center Hagenberg GmbH, Austria
András Förhécz, Multilogic Ltd., Hungary
Pierre-Yves Dumas, THALES, France

INTELLI Publicity Chairs

Frederick Ackers, Towson University, USA
Stephan Puls, Karlsruhe Institute of Technology, Germany
Paulo Couto, GECAD - ISEP, Portugal
Yuichi Kawai, Hosei University, Japan

INTELLI 2014 Technical Program Committee

Syed Sibte Raza Abidi, Dalhousie University - Halifax, Canada
Witold Abramowicz, The Poznan University of Economics, Poland
Michael Affenzeller, HeuristicLab, Austraia
Samir Aknine, Université Lyon 1, France
Zaher Al Aghbari, University of Sharjah, UAE
Firas B. Ismail Alnaimi, Universiti Tenaga Nasional, Malaysia
Ioannis Anagnostopoulos, University of Thessaly, Greece
Rachid Anane, Coventry University, UK
Andreas S. Andreou, Cyprus University of Technology - Limassol, Cyprus
Ngamnij Arch-int, Khon Kaen University, Thailand
Wudhichai Assawinchaichote, Mongkut's University of Technology -Bangkok, Thailand
Pradeep Atrey, University of Winnipeg, Canada
Paul Barom Jeon, Samsung Electronics, Korea
Daniela Barreiro Claro, Federal University of Bahia, Brazil
Rémi Bastide, Université Champollion, France
Carmelo J. A. Bastos-Filho, University of Pernambuco, Brazil
Bernhard Bauer, University of Augsburg, Germany
Barnabas Bede, DigiPen Institute of Technology - Redmond, USA
Noureddine Belkhatir, University of Grenoble, France
Orlando Belo, University of Minho, Portugal
Petr Berka, University of Economics, Prague, Czech Republic
Félix Biscarri, University of Seville, Spain
Luis Borges Gouveia, University Fernando Pessoa, Portugal
Abdenour Bouzouane, Université du Québec à Chicoutimi, Canada
José Braga de Vasconcelos, Universidade Atlântica, Portugal
Stefano Bromuri, University of Applied Sciences Western Switzerland, Switzerland
Rui Camacho, Universidade do Porto, Portugal
Luis M. Camarinha-Matos, New University of Lisbon, Portugal
Longbing Cao, University of Technology - Sydney, Australia
Sérgio Campello, Escola Politécnica de Pernambuco - UPE, Brazil
Jose Jesus Castro Sanchez, Universidad de Castilla-La Mancha - Ciudad Real, Spain
Marc Cavazza, University of Teesside - Middlesbrough, UK
Kit Yan Chan, Curtin University - Western Australia, Australia
Chin-Chen Chang, Feng Chia University, Taiwan, R. O. C.
Maiga Chang, Athabasca University, Canada
Yue-Shan Chang, National Taipei University, Taiwan
Naoufel Cheikhrouhou, Ecole Polytechnique Fédérale de Lausanne, Switzerland
Rung-Ching Chen, Chaoyang University of Technology, Taiwan
Li Cheng, BII/A*STAR, Singapore
Been-Chian Chien, National University of Tainan, Taiwan
Sunil Choenni, Ministry of Security and Justice, The Netherlands
Byung-Jae Choi, Daegu University, Korea

Sven Hartmann, Technische Universität Clausthal, Germany
Fumio Hattori, Ritsumeikan University - Kusatsu, Japan
Jessica Heesen, University of Tübingen, Germany
Pilar Herrero, Universidad Politecnica de Madrid, Spain
Benjamin Hirsch, Khalifa University - Abu Dhabi, United Arab Emirates
Didier Hoareau, University of La Réunion, France
Tetsuya Murai Hokkaido, University Sapporo, Japan
Wladyslaw Homenda, Warsaw University of Technology, Poland
Katsuhiro Honda, Osaka Prefecture University, Japan
Tzung-Pei Hong, National University of Kaohsiung, Taiwan
Samuelson W. Hong, Zhejiang University of Finance & Economics, China
Bin Hu, Birmingham City University, UK
Yo-Ping Huang, National Taipei University of Technology - Taipei, Taiwan
Germán Hurtado, University College Ghent & Ghent University, Belgium
Ming Huwi Horng, National PingTung Institute of Commerce, Taiwan
Carlos A. Iglesias, Universidad Politecnica de Madrid, Spain
Fodor János, Óbuda University – Budapest, Hungary
Jayadeva, Indian Institute of Technology - Delhi, India
Antonio Jimeno, NICTA - Melbourne, Australia
Yanguo Jing, London Metropolitan University, UK
Maria João Ferreira, Universidade Portucalense - Porto, Portugal
Janusz Kacprzyk, Polish Academy of Sciences, Poland
Epaminondas Kapetanios, University of Westminster - London, UK
Nikos Karacapilidis, University of Patras - Rion-Patras, Greece
Panagiotis Karras, Rutgers University, USA
Jung-jae Kim, Nanyang Technological University, Singapore
Sungshin Kim, Pusan National University- Busan, Korea
Abeer Khalid, International Islamic University Islamabad, Pakistan
Shubhalaxmi Kher, Arkansas State University, USA
Alexander Knapp, Universität Augsburg, Germany
Stephan Kopf, University of Mannheim, Germany
Sotiris Kotsiantis, University of Patras, Greece
Ondrej Krejcar, University of Hradec Kralove, Czech Republic
Natalia Kryvinska, University of Vienna, Austria
Satoshi Kurihara, Osaka University, Japan
K.P. Lam, University of Keele, UK
Antonio LaTorre, Universidad Politécnica de Madrid, Spain
Kennerd Laviers, Air Force Institute of Technology - Wright-Patterson
Frédéric Le Mouël, INRIA/INSA Lyon, France
Alain Léger, Orange - France Telecom R&D / University St Etienne - Betton, France
George Lekeas, City Universty – London, UK
Omar Lengerke, Autonomous University of Bucaramanga, Colombia
Carlos Leon, University of Seville, Spain
Haowei Liu, INTEL Corporation, USA

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

A Simulation Model for Transport in a Grid-based Manufacturing System

Leo van Moergestel, Erik Puik, Daniël Telgen,
Mathijs Kuijl, Bas Alblas and Jaap Koelewijn
Department of Computer science
HU Utrecht University of Applied Sciences
Utrecht, the Netherlands
Email: leo.vanmoergestel@hu.nl

John-Jules Meyer
Intelligent systems group
Utrecht University
Utrecht, the Netherlands
Email: J.J.C.Meyer@uu.nl

***Abstract*—Standard mass-production is a well-known manufacturing concept. To make small quantities or even single items of a product according to user specifications at an affordable price, alternative agile production paradigms should be investigated and developed. The system presented in this paper is based on a grid of cheap reconfigurable production units, called equiplets. A grid of these equiplets is capable to produce a variety of different products in parallel at an affordable price. The underlying agent-based software for this system is responsible for the agile manufacturing. An important aspect of this type of manufacturing is the transport of the products along the available equiplets. This transport of the products from equiplet to equiplet is quite different from standard production. Every product can have its own unique path along the equiplets. In this paper several topologies are discussed and investigated. Also, the planning and scheduling in relation to the transport constraints is subject of this study. Some possibilities of realization are discussed and simulations are used to generate results with the focus on efficiency and usability for different topologies and layouts of the grid and its internal transport system.**

***Keywords*-Multiagent-based manufacturing; Flexible transport.**

I. Introduction

In standard batch processing the movement of products is mostly based on a pipeline. Though batch processing is a very good solution for high volume production, it is not apt for agile manufacturing when different products at small quantities are to be produced by the production equipment. This paper describes an agile and flexible production system, where the production machines are placed in a grid. Products are not following a single path, but different paths can be used in parallel, leading to parallel production of different products. The grid arrangement of production machines reduces the average path when products move along their own possibly unique paths within the grid during the production. To move the products around during production, the ways the production machines are interconnected should be investigated to find an affordable and good solution. An important aspect will also be the amount of products in the grid during production, because too many products will result in failures in the scheduling of the production. The investigation about transport and the amount of products in the grid are the motivation and purpose of this paper. The goal is to investigate the effect of different interconnection possibilities to the average production path and to see how the grid behaves under load.

The work in this paper is based on our previous work. The design and implementation of the production platforms and the idea to build a production grid can be found in Puik [1]. In Moergestel [2] the idea of using agent technology as a software infrastructure is presented. Two types of agents play a major role in the production: a product agent, responsible for production of a product and an agent responsible for performing certain production steps on a production machine. Another publication by Moergestel [3] is dedicated the production scheduling for the grid production system.

In the next section of this paper related work will be discussed. Next, grid manufacturing will be explained in more detail, followed by a section about transport in the grid. After introducing the software tools built, the results are presented and discussed. Finally a conclusion where the results are summarized, will end the paper.

II. Related work

Using agent technology in industrial production is not new though still not widely accepted. Important work in this field has already been done. Paolucci and Sacile [4] give an extensive overview of what has been done in this field. Their work focuses on simulation as well as production scheduling and control [5]. The main purpose to use agents in [4] is agile production and making complex production tasks possible by using a multi-agent system. Agents are also introduced to deliver a flexible and scalable alternative for manufacturing execution systems (MES) for small production companies. The roles of the agents in this overview are quite diverse. In simulations agents play the role of active entities in the production. In production scheduling and control agents support or replace human operators. Agent technology is used in parts or subsystems of the manufacturing process. We on the contrary based the manufacturing process as a whole on agent technology. In our case a co-design of hardware and software was the basis.

Bussmann and Jennings [6][7] used an approach that compares to our approach. The system they describe introduced three types of agents, a workpiece agent, a machine agent and a switch agent. Some characteristics of their solutions are:

 ISBN: 978-1-61208-352-0

- The production system is a production line that is built for a certain product. This design is based on redundant production machinery and focuses on production availability and a minimum of downtime in the production process. Our system is a grid and is capable to produce many different products in parallel;
- The roles of the agents in this approach are different from our approach. The workpiece agent sends an invitation to bid for its current task to all machine agents. The machine agents issue bids to the workpiece agent. The workpiece agent chooses the best bid or tries again. In our system the negotiating is between the product agents, thus not disrupting the machine agents;
- They use a special infrastructure for the logistic subsystem, controlled by so called switch agents. Even though the practical implementation is akin to their solution, in our solution the service offered by the logistic subsystems can be considered as production steps offered by an equiplet and should be based on a more flexible transport mechanism.

So there are however important differences to our approach. The solution presented by Bussmann and Jenning has the characteristics of a production pipeline and is very useful as such, however it is not meant to be an agile multi-parallel production system as presented here.

Other authors focus on using agent technology as a solution to a specific problem in a production environment. The work of Xiang and Lee [8] presents a scheduling multiagent-based solution using swarm intelligence. This work uses negotiating between job-agents and machine-agents for equal distribution of tasks among machines. The implementation and a simulation of the performance is discussed. In our approach the negotiating is between product agents and load balancing is possible by encouraging product agents to use equiplets with a low load. We did not focus on a specific part of the production but we developed a complete production paradigm based on agent technology in combination with a production grid. This model is based on two types of agents and focuses on agile multiparallel production. There is a much stronger role of the product agent and a product log is produced per product. This product agent can also play an important role in the life-cycle of the product.

III. GRID MANUFACTURING

In grid production, manufacturing machines are placed in a grid topology. Every manufacturing machine offers one or more production steps and by combining a certain set of production steps, a product can be made. This means that when a product requires a given set of production steps and the grid has these steps available, the product can be made [1]. The software infrastructure that has been used in our grid, is agent-based. Agent technology opens the possibilities to let this grid operate and manufacture different kinds of products in parallel, provided that the required production steps are available [2]. The manufacturing machines that have been built in our research group are cheap and versatile. These machines are called equiplets and consist of a standardized frame and subsystem on which several different front-ends can be attached. The type of front-end specifies what production steps a certain equiplet can provide. This way every equiplet acts as a reconfigurable manufacturing system (RMS) [9]. An example of an equiplet front-end is a delta-robot. With this front-end, the equiplet is capable of pick and place actions. A computer vision system is part of the frontend. This way the equiplet can localise parts and check the final position they are put in. For a product to be made a sequence of production steps has to be done. More complex products need a tree of sequences, where every sequence ends in a half-product or part, needed for the end product. The equiplet is represented in software by a so-called equiplet agent. This agent advertises its capabilities as production steps to a blackboard that is available in a multiagent system where also so-called product agents live. A product agent is responsible for the manufacturing of a single product and knows what to do, the equiplet agents knows how to do it. A product agent selects a set of equiplets based on the production steps it needs and tries to match these steps with the steps advertised by the equiplets. The planning and scheduling of a product is an atomic action, done by the product agent in cooperation with the equiplet agent and takes seven steps [3]. Let us first assume that a single sequence of steps is needed.

1) From the list of production steps, build a set of equiplets offering these steps;
2) Ask equiplets about the feasibility and duration of the steps;
3) Indicate situations where consecutive steps on the same equiplet are possible;
4) Generate at most four paths along equiplets;
5) Calculate the paths along these equiplets;
6) Schedule the shortest product path using first-fit (take the first opportunity in time for a production step) and a scheduling scheme known as earliest deadline first (EDF) [3];
7) If the schedule fails, try the next shortest path;

For more complex products, consisting of a tree of sequences, the product agent spawns child agents, that are each responsible for a sequence. The parent agent is in control of its children and acts as a supervisor. It is also responsible for the last single sequence of the product. In Figure 1, the first two halfproducts are made using stepsequences $<\sigma_1, \sigma_2>$ and $<\sigma_3, \sigma_4>$. These sequences are taken care of by child agents, while the parent agent will complete the product by performing the step sequence $<\sigma_4, \sigma_7, \sigma_2, \sigma_1>$. Every product agent is

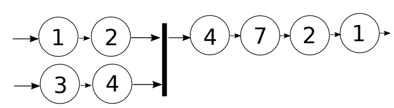


Figure 1: Manufacturing of a product consisting of two half-products

responsible for only one product to be made. The requests for products arrive at random. In the implementation we have made, a webinterface helps the end-user to design his specific product. At the moment all features are selected a product agent will be created. During manufacturing a product is guided by the product agent from equiplet to equiplet. This will in general be a random walk along the equiplets. This random walk is more efficient when the equiplets are in a grid arrangement against a line arrangement as used in batch processing.

IV. Transport in the grid

In the production grid, there is at least the stream of products to be made. Another stream might be the stream of raw material, components or half-products used as components. We will refer to this stream as the stream of components. These components could be stored inside the equiplets, but in that case there is still a stream of supply needed in case the locally stored components run short. This increases the logistic complexity of the grid model. In the next subsections, models will be introduced that alleviate the complexity by combining the stream of products with the stream of components.

A. *Buiding box model*

In the building box model, a tray is loaded with all the components to create the product. To maintain agility, this set of components can be different for every single product. Before entering the grid, the tray is filled by passing through a pipeline with devices providing the components. In this phase a building box is created that will be used by the grid to assemble the product. The equiplets in the grid are only used for assembling purposes. Figure 2 shows the setup.

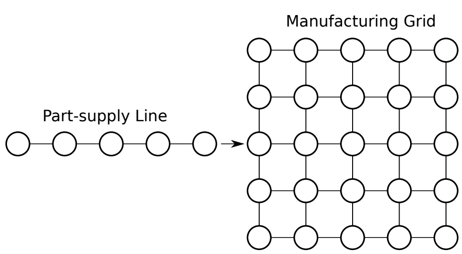


Figure 2: Production system with supply pipeline

A problem with the previous setup is the fact that more complex products should be built by combining subparts that should be constructed first. In the previously presented setup all parts needed for the construction of the subparts should be collected in the building box, making the assembling process more complicated. Another disadvantage of putting all components for all subparts together in a building box is that this slows down the production time, because normally subparts can be made in parallel. A solution is shown in the setup of Figure 3. Subparts can be made in parallel and are input to the supply-line that eventually could be combined with the original supply-line. The next refinement of the system is presented in Figure 4. Here a set of special testnodes has been added to the system. These nodes are actually also equiplets, but these equiplets have a front-end that makes them suited for testing and inspecting final products as well as subparts that should be used for more complicated products. A test can also result in a reject and this will also inform the product agent about the failure. If the product agent is a child agent constructing a subpart, it should consult the parent agent if a retry should be done. In case it is the product agent for the final product, it should ask its maker what to do.

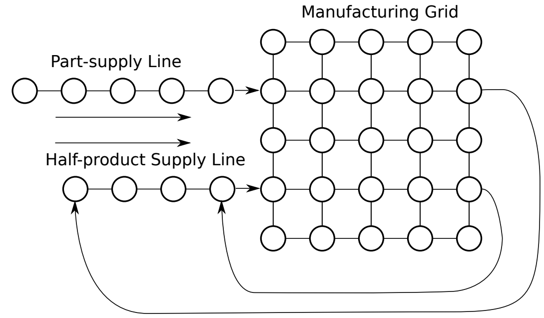


Figure 3: Production system with loops

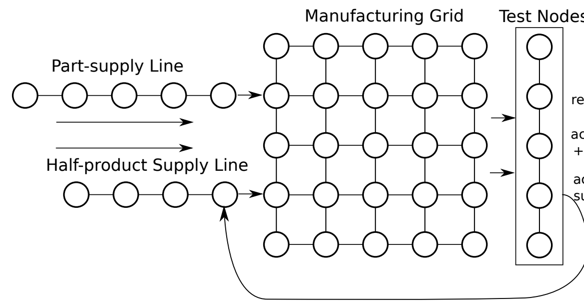


Figure 4: Production system with loops

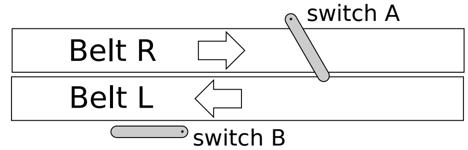


Figure 5: Bidirectional conveyor belt with switches

B. *Conveyor belt-based systems*

A conveyor belt is a common device to transport material. Several types are in use in the industry. Without going into detail, some kind of classification will be presented here:

- belts for continuous transport in one direction;
- belts with stepwise transport from station to station. These types of belts can be used in batch environments, where every step takes the same amount of time and the object should be at rest when a production step is executed;
- belts with transport is two directions. This can also be realised by using two one direction belts, working in opposite direction.

In Bussmann [6], an agent-based production system is built using transport belts in two directions where a switch mechanism can move a product from one belt to another. A special switch-agent is controlling the switches and thus controlling the flow of a product along the production machines. In Figure 5 this solution is shown. Switch A is activated and will shift products from belt R to the belt L that will move it to the left. This concept fits well in the system developed by Bussmann, because that system is actually a batch-oriented system. In a grid the use of conveyor belts might be considered, but for agile

 ISBN: 978-1-61208-352-0

transport several problems arise, giving rise to complicated solutions:

- should the direction in the grid consist of one-way paths or should be chosen for bidirectional transport?
- a product should be removed from the moving belt during the execution of a production step. A stepwise transport is inadequate, because of the fact that production steps can have different execution times in our agile model. This removal could be done by a switch mechanism as used by Bussmann, but every equiplet should also have it own switch-unit to move the product back to the belt.
- because the grid does not have a line structure for reasons explained in the first part of this chapter, a lot of crossings should be implemented. These crossings can also be realised with conveyor belt techniques, but it will make the transport system as a whole expensive and perhaps more error-prone.

C. Autonomous transport

An alternative for conveyor belts is the use of automatic guided vehicles (AGV). An AGV is a mobile robot that follows certain given routes on the floor or uses vision, ultrasonic sonar or lasers to navigate. These AGVs are already used in industry mostly for transport, but they are also used as moving assembly platforms. This last application is just what is needed in the agile manufacturing grid. The AGV solution used to be expensive compared to conveyor belts but some remarks should be made about that:

- These AGV offer a very flexible way for transport that fits better in non-pipeline situations;
- Low cost AGV platforms are now available;
- From the product agent view, an AGV is like an equiplet, offering the possibility to move from A to B.
- A conveyor-belt solution that fits the requirements needed in grid productions will turn out to be a complicated and expensive system due to the requirements for flexible transport.

In the grid a set of these AGVs will transport the product between equiplets and will be directed to the next destination by product agents.

1) AGV system components: An AGV itself is a driverless mobile robot platform or vehicle. This AGV is mostly a battery-powered system. To use an AGV, a travel path should be available. When more then one AGV is used on the travel path. A control system should manage the traffic and prevent collisions between the AGVs or prevent deadlock situations. The control system can be centralised or decentralised.

2) AGV navigation: There are plenty ways in which navigation of AGVs has been implemented. The first division in techniques can be made, based on whether the travel path itself is specially prepared to be used by AGVs. This can be done by:

- putting wires in the path the AGV can sense and follow;
- using magnetic tape to guide the AGV;
- using coloured paths, by using adhesive tape on the path to direct the AGV;
- using transponders, so the AGV can localise itself.

The second type of AGV does not require a specially prepared path. In that case navigation is done by using:

- laser range-finders
- ultrasonic distance sensors
- vision systems

Though it might look as if the decision for using AGVs has already been made, further research should be done to see what the efficiency will be for several implementations. This will be the subject of the next two sections.

V. Software Tools

Two simulation software packages have been built. A simulation of the scheduling for production and a simulation for the path planning. The path planning tool will be used to calculate the efficiency for different transport interconnections. The scheduling tool will be used to calculate the number of active product agents within the grid. This number is important, because it will tell how many products should be temporally stored, waiting for the next production step to be executed.

A. Path planning simulation software

A path planning tool has been built, to calculate a path a certain product has to follow along the equiplets. The Dijkstra path algorithm has been used [10]. The tool can work on different grid transport patterns. This tool will be used to study several possible grid topologies. A screen-shot of the graphical user interface of the tool is shown in Figure 6. Several different topologies and interconnections can be chosen by clicking the appropriate fields in the GUI. The average transport path for all nodes is one of the results of this simulation.

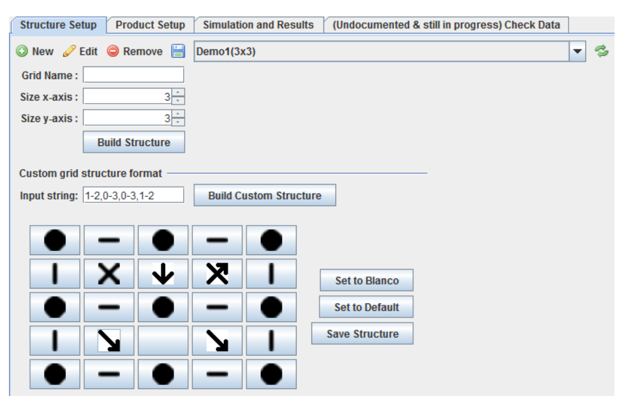


Figure 6: Path planning GUI

B. *Scheduling simulation software*

The software for the scheduling simulations consists of two parts. One part is a command-line tool that is driven by a production scenario of a collection of product agents, each having their own release time, deadline and set of production steps. This production scenario is a human readable XML-file. The second part is a GUI for visualisation of the scheduling system. In Figure 7 a screen shot of this visualisation tool is shown.

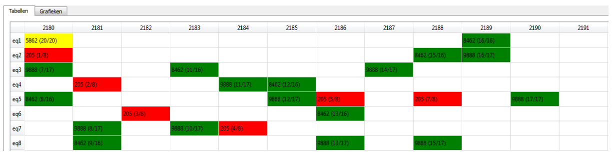


Figure 7: GUI of the scheduling simulator

VI. RESULTS

To calculate the average pathlength in the grid for different paths, several structures have been investigated. Some of these structures were chosen to fit conveyor belt solutions of some type. All structures will also fit within the AGV-based solution.

- A fully connected grid. where all paths are bidirectional paths as in Figure 8.
- A grid where all paths are bidirectional, but this design has removed the crossings as in Figure 9. This structure could be implemented by conveyor belts in combination with switches;
- A structure with five unidirectional paths and two bidirectional paths as in Figure 10. This structure is also a possible implementation with conveyor belts;
- A structure with bidirectional paths combined in a single backbone as in Figure 11;
- A structure with five bidirectional paths and two unidirectional paths as in Figure 12;
- A fully connected grid, but now with half of the paths unidirectional as in Figure 13.

For all these structures the average path is the result from a simulation of 1000 product agents, all having a random walk within the grid. Each product agent has an also random set of equiplets it has to visit ranging from 2 to 50 equiplets per product agent. Every path or hop between adjacent nodes is considered to be one unit length. If the paths have no crossings, a conveyor belt might be used, because crossing belts will result in a more complex system. All structures can also be implemented with AGVs. For some structures the average path can also easily be calculated and the results of these exact calculations are within 1% of the simulation results.

The results of the simulation are given in a table and also plotted as a histogram in Figure14. In Table I, a second outcome from the simulation is also shown. This is the percentage of agents that could find an alternative path of the same length. This result is of interest when in a traffic control implementation, alternative paths become important.

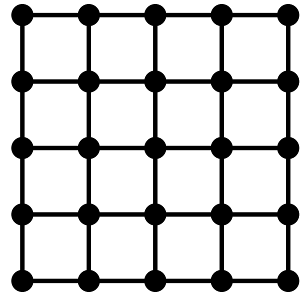


Figure 8: Standard fully connected grid

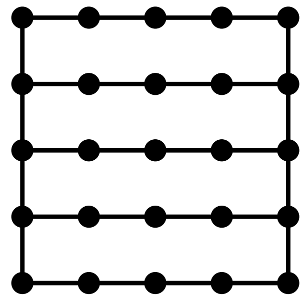


Figure 9: grid with bidirectional lanes and bidirectiona backbone lanes

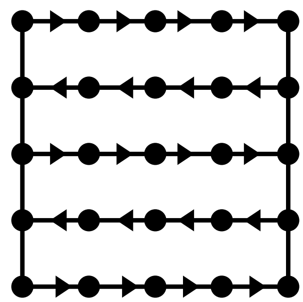


Figure 10: Grid with unidirectional lanes and bidirectional backbone lanes

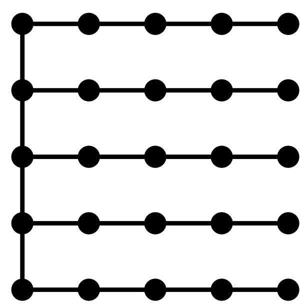


Figure 11: E-shaped connection, with bidirectional lanes

As could be expected, the best result is achieved in the fully connected grid with bidirectional paths. Changing the grid to an almost identical structure of Figure 13 with unidirectional paths, results in only a small penalty. This structure could also be useful in an AGV-based transport system, reducing collision problems because of the one-way paths used. Both structures also offer a relative high percentage of alternative paths, that could also be useful in an AGV-based system. The structures that fit a conveyor belt solution show a path length that is

 ISBN: 978-1-61208-352-0

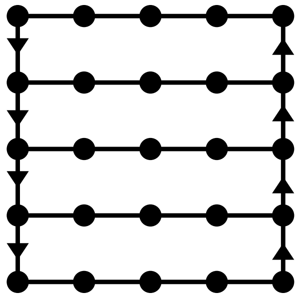


Figure 12: Bidirectional lanes with unidirectional backbones

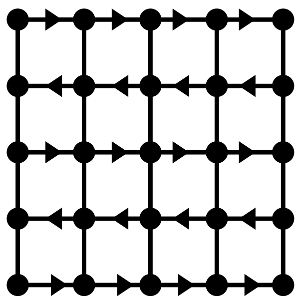


Figure 13: Fully connected grid with unidirectional lanes

TABLE I: Results of the simulatiom

Structure	1	2	3	4	5	6
Average path	3.2	3.9	6.4	5.1	6.0	3.6
% Alternatives	60	16.7	8.4	0	0	27

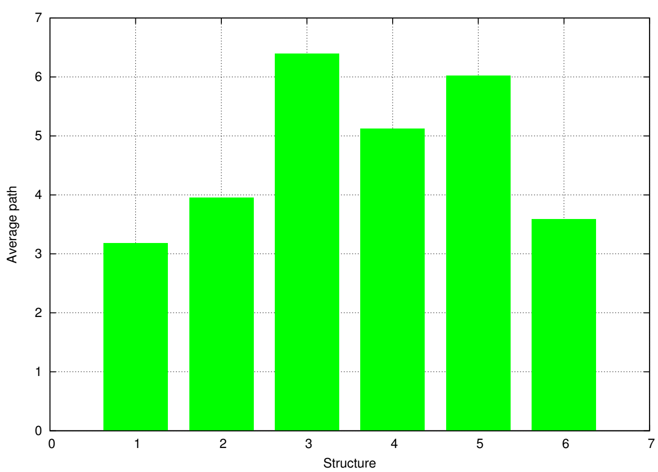


Figure 14: Simulation results for different structures

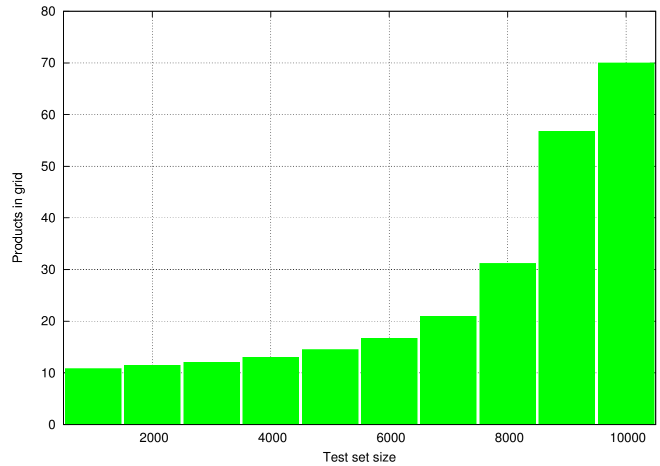


Figure 15: Simulation Results for different sizes op product sets

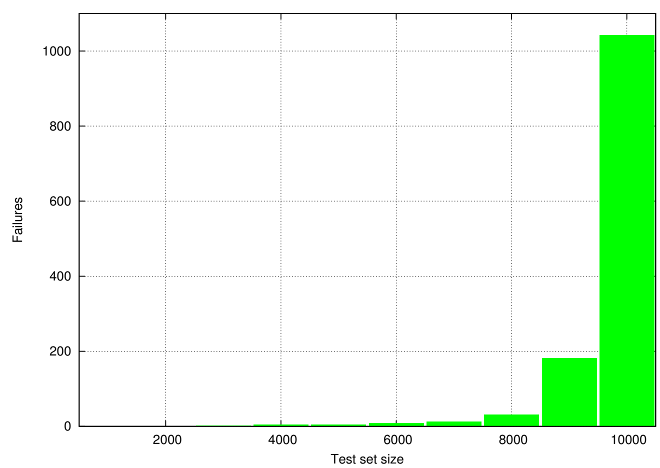


Figure 16: Simulation Results for different sizes op product sets

considerably higher.

The next results were generated using the scheduling tool. This tool was used in earlier research [3] to discover the scheduling approach to be used. Earliest deadline first (EDF) turned out to be a good choice. In Figure 15 the average number of product agents in a manufacturing grid consisting of 10 equiplets is shown for different sizes of test sets. Every product agent has a random number of equiplets to visit ranging from 1 to 20. Also, the time window between release time and deadline is random between 1 to 20 times the total production time of a product. The number of timesteps is 10000 and the duration of a production step is 1 timestep. For a set with 10000 product agents, the grid is actually overloaded as can be seen in Figure 16, where the number of scheduling failures is over 1000. Another simulation shows the same effect. This simulation is based on one scenario with a linear increasing amount of product agents in time as shown in Figure 17. In this graph, a product is considered active in the grid between its release time and its deadline. The actual number of products in the grid is shown in Figure 18. When we look at the actual number of active products in the grid, the resulting graph shows an remarkable shape. In the beginning, the actual number is even less than the number plotted in the graph of Figure 17. This is due to the fact that in a grid that is only used by a small amount of products, every product will be finished far before its deadline. A finished product is not considered active in the grid any more. However, at a certain point there is a steep increase in the number of products and the graph saturates at the same level of 70 products as shown in Figure 15 for a test set of 10000. The number of rejected products due to a failing scheduling will increase. This also means that overloading the grid will generate many

 ISBN: 978-1-61208-352-0

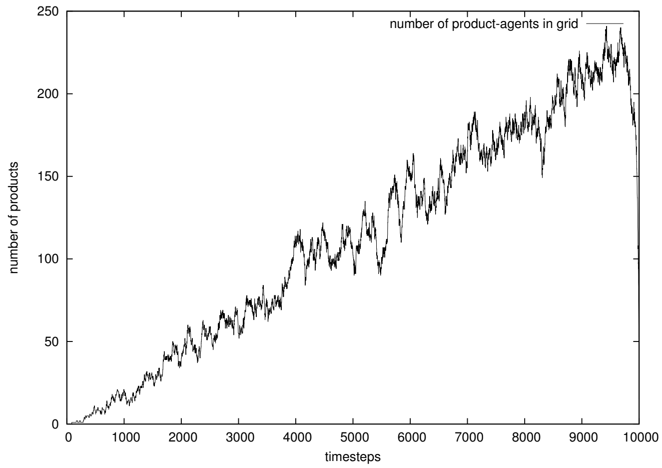


Figure 17: Increasing number of active products in the grid

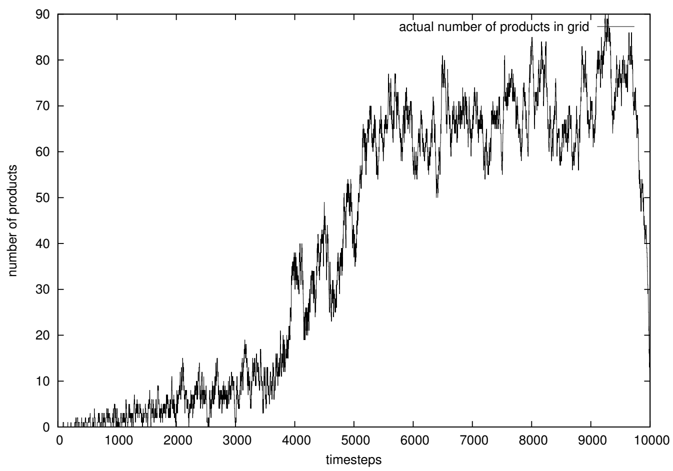


Figure 18: Number of products in the grid

active products that should be stored somewhere, because in this given situation, only 10 products can be handled by an equiplet.

VII. Conclusion

For the agile grid-based and agent-based manufacturing the buildingbox as well as the AGV-based system offer advantages:

- By using a building box, the transport of parts to the assembling machines (equiplets) is combined with the transport of the product to be made. It will not happen that a part is not available during manufacturing;
- Because the product as well as it parts use one particular AGV during the production, there is never a competition for AGV during the manufacturing process;
- An AGV can use the full possibility and advantage of the grid-based system being a compact design resulting in short average paths;
- The product agent knows which equiplets it should visit and thus can use the AGV in the same way as an equiplet. The product agent can instruct the AGV agent to bring it to the next equiplet in the same way as it can instruct an equiplet agent to perform a production step;
- An AGV can bring the production platform exact to the right position for the equiplet and can even add extra movement in the X-Y plane or make a rotation around the Z-axis;
- If an AGV fails during production the problem can be isolated and other AGVs can continue to work. In a conveyor belt system a failing conveyor might block the whole production process.

There are also some disadvantages:

- There should be a provision for charging the battery of the AGV;
- Simulations show that the amount of agents in the grid shows a strong increase in a grid that is loaded over 80%. This will result in a lot of AGVs in the grid leading to traffic jam;
- Only products that fit within the building box manufacturing model can be made.

Agent-based grid manufacturing is a feasible solution for agile manufacturing. Some important aspects of this manufacturing paradigm have been discussed here. Transport can be AGV-based provided that the load of the grid should be kept under 80% to overcome the temporary storage requirements.

References

[1] E. Puik and L. v. Moergestel, "Agile multi-parallel micro manufacturing using a grid of equiplets," Proceedings of the International Precision Assembly Seminar (IPAS 2010), 2010, pp. 271–282.

[2] L. v. Moergestel, J.-J. Meyer, E. Puik, and D. Telgen, "Decentralized autonomous-agent-based infrastructure for agile multiparallel manufacturing," Proceedings of the International Symposium on Autonomous Distributed Systems (ISADS 2011) Kobe, Japan, 2011, pp. 281–288.

[3] L. v. Moergestel, J.-J. Meyer, E. Puik, and D. Telgen, "Production scheduling in an agile agent-based production grid," Proceedings of the Intelligent Agent Technology (IAT 2012), 2012, pp. 293–298.

[4] M. Paolucci and R. Sacile, Agent-based manufacturing and control systems : new agile manufacturing solutions for achieving peak performance. Boca Raton, Fla.: CRC Press, 2005.

[5] E. Montaldo, R. Sacile, M. Coccoli, M. Paolucci, and A. Boccalatte, "Agent-based enhanced workflow in manufacturing information systems: the makeit approach," J. Computing Inf. Technol., vol. 10, no. 4, 2002, pp. 303–316.

[6] S. Bussmann, N. Jennings, and M. Wooldridge, Multiagent Systems for Manufacturing Control. Berlin Heidelberg: Springer-Verlag, 2004.

[7] N. Jennings and S. Bussmann, "Agent-based control system," IEEE Control Systems Magazine, vol. 23, no. 3, 2003, pp. 61–74.

[8] W. Xiang and H. Lee, "Ant colony intelligence in multi-agent dynamic manafacturing scheduling," Engineering Applications of Artificial Intelligence, vol. 16, no. 4, 2008, pp. 335–348.

[9] Z. M. Bi, S. Y. T. Lang, W. Shen, and L. Wang, "Reconfigurable manufacturing systems: the state of the art," International Journal of Production Research, vol. 46, no. 4, 2008, pp. 599–620.

[10] M. Sniedovich, "Dijkstras algorithm revisited: the dynamic programming connexion," Control and Cybernetics, vol. 35, no. 3, 2006, pp. 599–620.

Triggering Solar-Powered Vehicle Activated Signs using Self Organising Maps with K-means

Diala Jomaa, Siril Yella, Mark Dougherty
Department of Computer Engineering, Dalarna University
78170 Borlänge, Sweden
djo@du.se, sye@du.se, mdo@du.se

Abstract— Solar-powered vehicle activated signs (VAS) are speed warning signs powered by batteries that are recharged by solar panels. These signs are more desirable than other active warning signs due to the low cost of installation and the minimal maintenance requirements. However, one problem that can affect a solar-powered VAS is the limited power capacity available to keep the sign operational. In order to be able to operate the sign more efficiently, it is proposed that the sign be appropriately triggered by taking into account the prevalent conditions. Triggering the sign depends on many factors such as the prevailing speed limit, road geometry, traffic behaviour, the weather and the number of hours of daylight. The main goal of this paper is therefore to develop an intelligent algorithm that would help optimize the trigger point to achieve the best compromise between speed reduction and power consumption. Data have been systematically collected whereby vehicle speed data were gathered whilst varying the value of the trigger speed threshold. A two stage algorithm is then utilized to extract the trigger speed value. Initially the algorithm employs a Self-Organising Map (SOM), to effectively visualize and explore the properties of the data that is then clustered in the second stage using K-means clustering method. Preliminary results achieved in the study indicate that using a SOM in conjunction with K-means method is found to perform well as opposed to direct clustering of the data by K-means alone. Using a SOM in the current case helped the algorithm determine the number of clusters in the data set, which is a frequent problem in data clustering.

Keywords: Solar-powered vehicle activated signs; Self Organising Maps; K-means clustering; Trigger speed

I. INTRODUCTION

An excessive or inappropriate speed is often a reason for traffic fatalities. Therefore an important consideration for traffic municipalities is to reduce speeding by either modifying the roadway infrastructure or introducing additional and more effective signage. Modifying road infrastructure is more costly than deploying additional signs. Therefore, a range of road safety signs has been developed and deployed to encourage drivers to adapt to the speed limit or to warn drivers when they are approaching a hazard. Solar vehicle activated signs (VAS) are one type of signs that are widely used on roadways. Typically, solar VAS are speed warning sign powered by batteries that are recharged by solar panels. These signs are more desirable than other simple battery driven VAS due to their low cost of installation. A main source of the reduction in cost is due to the fact that no external power supply is needed. These signs usually consist of radar that is mounted inside the sign in order to detect vehicles and measure their speed. The sign displays a message when vehicle speed exceeds a pre-set threshold, which is called the trigger speed. The trigger speed is usually set to a constant value, which is often equal, or relative, to the speed limit on a particular segment of road. Earlier studies reviewing the effectiveness of variable message signs or vehicle activated signs have been reported by [1]. Such studies have reviewed relevant work published between 2000 and 2005 and have mainly investigated the influence of VMS on human behaviour. This study is an update to the earlier studies in that this study includes data from 2006-2009 [2]. The prior studies showed that these signs have a significant impact on driver behaviour, traffic safety and traffic efficiency. In most cases, the signs have yielded reductions in the mean speed and in speed variation as well as in longer headway. However, most of the experiments were performed with the signs set to a certain static configuration under specific conditions. Since some of the aforementioned factors are dynamic in nature, it is felt that the earlier researchers did not consider the aspects of sign configuration carefully enough. The previous studies lack a clear statement describing the relationship between the trigger value and its consequences under different conditions [3]. Efficiently setting up the radar speed threshold helps prevent the battery from running out of power. Previous authors have reported different strategies for calculating the appropriate trigger speed. In one reported experiment the trigger speed was set at 10% over the speed limit plus an additional 2mph, i.e., in a 30mph speed limit the trigger speed would be set at 35mph) [4]. The trigger speed was set at the 50^{th} percentile of the speed that was detected prior to the installation of the VAS. This was intended to target half of the drivers. In other previous studies, the trigger speeds were set at between the 75th and 81th percentile speeds [5]. In Mattox et al., the predetermined trigger speed was set to the posted speed limit with a 3- mph buffer [6]. The method that is used predominantly in the United Kingdom is one in which the trigger speed is set to the 85th percentile of the average speed that is measured before installation.

However, solar VAS are often challenged by a limitation is the capacity of the power supply that is required for the sign to remain operational. In order to be able to operate the sign more efficiently, it is proposed that the sign trigger should take into account the prevalent conditions. Triggering the sign depends on many factors such as the prevailing speed limit, road geometry, traffic behaviour and number of hours of daylight (more daylight implies more solar power available, but on the other hand the sign needs to shine brighter during daylight, which implies higher power consumption). As a consequence of the lack of power capacity, the sign may be triggered with a high value but it should preserve the impact of the sign on vehicle speed reduction. To determine the optimal trigger speed in order to minimize power consumption while simultaneously maximizing vehicle speed reduction is a nontrivial problem. Thus, the optimal trigger speed will be accomplished by first collecting traffic data using various trigger speeds, pre-process the data and extract the trigger speeds that reveal the information and relationship hidden in these data for example, the relationship between the time of day and the traffic conditions. The objective of this paper is to therefore develop an intelligent algorithm that searches for the optimal trigger speed, which yields the best compromise between reducing both vehicle speeds and the power consumption of the sign. The algorithm is mainly done by combining two clustering techniques, Self-Organising Map (SOM) and K-means. In this algorithm, traffic data will be first clustered by a SOM in order to effectively visualize, explore the properties of the data and determine the preliminary number of clusters as well as determine the centre of each cluster. After using the SOM, the clusters are refined by using K-means. The remainder of this paper is organized as follows. Section 2 details the experimental design and discusses data acquisition. Section 3 presents issues relevant to solar power consumption for the sign. The developed algorithm and its trigger speed are described extensively in sections 4 and 5. The results and discussion are reported in section 6. Lastly concluding remarks will be presented.

II. SOLAR POWER CONSUMPTION

Solar powered signs are allow for signs to be easily installed in locations that are far away from the power grid. They are designed to run all year round, particularly running more in the winter when there is less light. The radar is always on and this draws somewhere in the region of 40-50mA. However, the only other power draw is the controller and 3 LEDs on the rear of the sign, which require about 20mA in total. The total current from the 12V battery is then around 70mA. When the sign faced is active and at full brightness, the current consumption is 1.8A. This reduces in steps as the ambient light reduces right down to under 0.5A. The modem that is fitted to this unit is also drawing current all of the time this is in the region of 50mA and has an impact on battery life. The current taken by the unit depends on the quality of the network signal that is available at the site and the amount of data that is being moved, so it is not an easy sum to calculate. Therefore there is a need to consume the energy efficiently in order to ensure that the VAS has an adequate amount of power. The established sign runs on 40W solar panels with 35 Ah batteries. In this study, the energy consumption is calculated as follows. If the vehicle speeds exceed the proposed threshold speed then:

- Calculate the length of flash f for each vehicle speed
- Sum all flashes
- Calculate energy consumption at full brightness in Ah

III. EXPERIMENT DESIGN AND DATA ACQUISITION

Solar Traffic agencies have a specific policy for the use of interactive signs such as VAS. These policies stipulate where the sign can be placed and the possible roadways for which they are suitable. Such signs are mostly placed at or near speed limit changes or at sites where a high collision rate exists. Sites near junctions and pedestrian crossings are normally avoided, as vehicle speeds are generally reduced in these areas [7]. In this study, two test sites were selected in Borlänge, Sweden. The first site is referred to as the Korsgård test site. The Korsgård test site is located between the Tuna and Hugo Hedström roadways. The second site is referred to as the Mjälga test site. Note that traffic flows in both directions and the posted speed limit at the test sites is 40 km/hr. Furthermore both sites are located in rural area and are notorious for speeding. At both sites, two radars (radar 1 and radar 2) were used in order to study the reduction of vehicle speed before and after triggering the sign. Radar 1 was positioned 100 meters before the VAS and radar 2 was positioned in line with the VAS. The VAS is triggered at 100 m in distance before the location of the sign. The sign used in the current study is a typical solar powered vehicle activated sign (VAS). The sign displays two warning messages in succession. The first is a reminder of the posted speed limit, which is 40km/hr, which is followed by a "SÄNK FARTEN" (reduce speed) message. Typically the messages are displayed only when the vehicle speed exceeds the trigger speed. The sign is equipped with radar and a data logger to detect and record vehicle speed. The sign is also equipped with a general packet radio service (GPRS) modem to facilitate communication to the radar and for authorized users to download and upload data. Note that it has been possible to alter the radar settings remotely. Such a setup has facilitated alteration of the trigger speed, thereby permitting the study to investigate the effect of different trigger speeds on various driving speeds. Upon request the data stored in the collection module is uploaded to a web server for the user to download.

At both sites, the data were collected 24-hours a day. At the Korsgård test site, data collection were done from 1 September 2012 through 31 December 2012 and at the

Mjälga test site, data collection were established between from 1 May 2013 and 1 August 2013. The data collected consists of records about vehicle speed, the direction the vehicle is traveling, the length of vehicle, a time stamp and a time gap and the date. Only the vehicle speed, direction and the time of day were used in this study.

IV. A TWO STAGE CLUSTERING APPROACH USING SELF-ORGANISING MAP WITH K-MEANS CLUSTERING

Clustering is an unsupervised classification of observations, data items, or objects into different groups or clusters so that the data in each cluster share the same properties [8]. Grouping is usually done on the basis of similarities or distances without any assumptions concerning the number of groups that should be pertained to the data. There are various methods that are used for cluster analysis. Self-Organising Maps (SOM) and K-means are two methods that are commonly applied in several fields such as marketing, pattern recognition and traffic analysis. However, the performance of these methods can differ depending of the characteristics of the data, for example the size of the data, the number of clusters, and the type of data. Nevertheless, none of these methods outperforms the others in all data conditions. For instance, SOM is sensitive to the size of the data set; its speed of convergence is slower than K-means [9]. K-means is a simple and fast algorithm but it is very sensitive to the selection of the initial number of clusters [10]. In this paper, a Self-Organizing Map (SOM) was initially used to visualize, and understand data properties such as the number of prevalent clusters. Preliminary identification of the number of clusters is deemed useful because it enables the researcher to decide the number of clusters (K) while implementing the K-means algorithm [11].

A. Khohonen self-organising maps

The Self-organising map (SOM) is an artificial neural network that learns the properties of data via an unsupervised learning algorithm [12]. It consists of two layers of artificial neurons; an input layer and an output layer. Every input neuron is connected to every output neuron by a weighting value. The Euclidian distance is calculated between the input vector and the incoming weighted vector for each output. The output neuron with the smallest distance is declared as the winner and its weights modified to be closer to the input vector. In fact SOM is an iterative process where the connections' weights are modified according to the following equations (1) and (2) [13, 14].

$$\mathrm{w(t+1) = w(t) + h(t)(x(t) - w(t))} \quad (1)$$

$$\mathrm{h(t) = \alpha.\, e^{-\frac{d^2}{2\sigma^2(t)}}} \quad (2)$$

Where w(t) is the connection weight at time t,
x(t) is the input vector
h(t) is the neighbourhood function,
α is the learning rate,
d is the Euclidian distance between the winning unit and the current unit,
σ is the neighbourhood width parameter

B. K-means clustering

The k-means clustering is an optimization clustering algorithm where clusters are formed by optimizing some measure of cluster goodness. The centre of the cluster k is the mean of the data items within the cluster. The K-means algorithm proceeds by first randomly selecting k of the items where each selection is done by partitioning data items into k initial clusters. Each item is assigned to the cluster to which it is the most similar, based on the distance between the item and the cluster mean. It then computes the new mean for each cluster and assigns the new mean as the new cluster centre. This process iterates until a stopping condition is reached.

In this paper a two stage clustering algorithm is applied as follows:

a) Data pre-processing: The data is initially pre-processed by simple filtering. In this case the filtering consists based on the direction the vehicles are traveling. Vehicles traveling in the same direction are filtered into the same group. Later, the data is grouped into two classes: cars and trucks. Motorcycles and long trucks are excluded from this study by assuming that their speed is low.

b) Feature extraction: The main features considered in this study are: time mean speed μ, traffic flow q and the standard deviation of speed σ. Assuming that all of the vehicles are moving with v km/hr. The number of vehicles counted at a certain point in one hour is the traffic flow q. Time mean speed is the average of spot speed $\mathrm{v_i}$ or simply the average of n vehicles passing a point during a certain period of time [15]. Time mean speed μ is given by equation (3):

$$\mu = 1/n \sum_{i=1}^{n} v_i \quad (3)$$

Speed standard deviation is approximately the square root of the sum of the squares for the difference between each vehicle speed v_i and the mean speed μ [16].
Speed standard deviation is given by equation (4):

$$\sigma = \sqrt{1/n^2 \sum_{i=1}^{n} (v_i - \mu)} \quad (4)$$

 ISBN: 978-1-61208-352-0

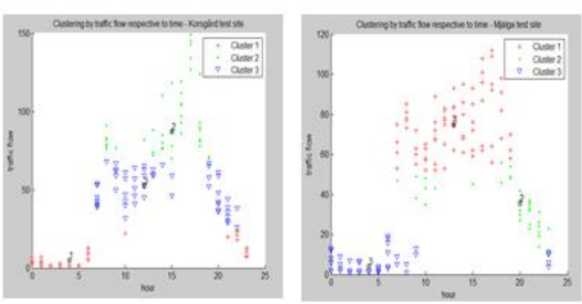
c) The SOM clustering stage: Obtain the initial number of clusters by SOM. The input vectors assigned to the SOM input neurons consist of 3 dimensions; time mean speed, traffic flow and standard deviation. In this experiment, a 24 by 3 layer of neurons is used to classify the input into 3 clusters. This means a layer of 72 neurons spread out in a 24 by 3 grid. After training the network with 100 iterations, the map is well distributed with regard to the input space.

d) The K-means clustering stage: Refine the cluster centroid by K-means. The number of clusters and the cluster centres obtained from SOM can be used as the initial input of the K-means algorithm.

e) Trigger speed setting: According to the clustering results obtained from the K-means, the trigger speed is considered as the median speed of the cluster.

V. RESULTS AND ANALYSIS

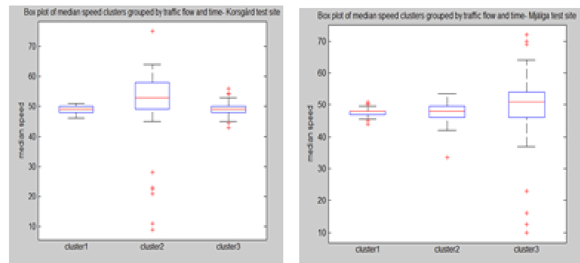
The two-stage clustering algorithm was applied to the experimental data obtained from both test sites; Korsgård and Mjälga. A choice of K=3 is based on the initial clustering done by SOM. Figure 1 (a) and (b) show how the K-means partitioned the traffic flow into 3 clusters at both the Korsgård and Mjälga test sites, respectively. This means grouping the time of day into three clusters that correspond to the number of vehicles passing the sign. At the Korsgård test site, the traffic flow is low in cluster 1, average in cluster 3, high in cluster 2, however at the Mjälga test site the traffic flow is low in cluster 3, average in cluster 2, and high in cluster 1. In fact, the partition of the time of day for the traffic flow is not the same in both sites.



(a) (b)

Figure 1. Clustering traffic flow respective to time; (a) at the Korsgård test site; (b) at the Mjälga test site

The main idea in using K-means clustering is to retrieve the median speed centroid, which is the expected trigger speed threshold. Fig. 2 (a) and (b) shows a box plot of the median speed for the three clusters at the Korsgård test site and at the Mjälga test site, respectively. The central mark in this box plot is the median, the edges of the box are the 25th and 75th percentiles and outliers are plotted individually. The whiskers in Cluster 2 at the Korsgård site and in cluster 3 at the Mjälga site are more extended than other clusters due to a larger deviation between median speeds. In fact when the traffic flow is high at the Korsgård test site, the median speed is high (cluster 2) but when the traffic is low at the Mjälga site, the median speed is high (cluster 3). Bear in mind, that both test sites have a speed limit of 40 km/hr.



(a) (b)

Figure 2. Box plot of median speed for the three clusters; (a) at the Korsgård test site; (b) at the Mjälga test site

For the purpose for this study, the 24 hours of the day were grouped into three time period clusters. The time periods obtained from the clustering algorithm ensured that the trigger speed is applied to the entire clusters, rather to the individual hours. The trigger speeds are considered as the centroid median speeds for the clusters obtained from the experiment. Tables 1 and 2 present the centroid median speed and the corresponding standard deviation of all of the clusters from the mean speed at the Korsgård and Mjälga test sites, respectively.

TABLE 1. THE TIME PERIOD RANGE, THE CENTROID'S MEDIAN SPEED AND THE CENTROID STANDARD DEVIATION FOR EACH CLUSTER AT THE KORSGÅRD TEST SITE

Clusters number (K)	Cluster's time period range	Centroid median speed	Centroid Standard deviation
1	{07,09,10,11,12,19,20,21}	49	09
2	{00,01,02,03,04,05,06,22,23}	51	13
3	{08,13,14,15,16,17,18}	49	08

TABLE 2. THE TIME PERIOD RANGE, THE CENTROID'S MEDIAN SPEED AND THE CENTROID STANDARD DEVIATION FOR EACH CLUSTER AT THE MJÄLGA TEST SITE

Clusters number (K)	Cluster's time period range	Centroid median speed	Centroid standard deviation
1	{00,01,02,03,04,05,06,22,23}	51	13
2	{07,09,10,20,21}	49	10
3	{08,11,12,13,14,15,16,17,18,19}	48	09

Both sites provide nearly identical results for the centroid median speed and the centroid standard deviation. However, the partitioning of the day is not the same in all clusters. The partitions are the same at night-time but it differs during the daytime.

VI. VALIDATION AND ENERGY CONSUMPTION

Validating the data is not a straightforward process. There are several cluster validation techniques. One of these techniques which were initially proposed by Hauser and Schere, is to break down the data into subsets and then try to reach the same clusters from the original data [17]. In this study, the validation is done by comparing the effect of the sign that was triggered with the clustering model to the effect of the sign that was triggered with other static trigger speeds. The static trigger speeds are obtained from the 15^{th}, 50^{th} and 85^{th} percentiles of the vehicle speeds. The 15^{th} percentile speed is the speed up to which 15% of vehicles travel. For example if the 15^{th} percentile speed was 100 km/hr then 15% of the traffic would travel at 100 km/hr or lower. The same definition applies to the 50^{th} and 85^{th} percentile speeds respectively. Bear in mind that 15th, 50th and 85th percentiles are reflecting a vehicle's speed at the specific test sites (as measured before the experiment began). To limit the effect to only weekdays, the evaluations were based only on the data that were collected on three Mondays. Table 3 shows the various trigger speeds applied at both test sites.

TABLE 3. THE VARIOUS TRIGGER SPEED (KM/H) AT THE TEST SITES

Korsgård test site		**Mjälga test site**	
Trigger speed basis	***Trigger speed (TS)***	***Trigger speed basis***	***Trigger speed (TS)***
15th percentile	42	15th percentile	46
50th percentile	47	50th percentile	49
85th percentile	52	85th percentile	50
Clustering model	52-47	Clustering model	52-49

The effect of the sign is done by calculating:

- The variation in the mean speed of a vehicle travelling before triggering the sign as well as after triggering the sign.
- Energy consumption

Fig. 3 and Fig. 4 show the energy consumed on the clustering model is actually near the 50^{th} percentile and larger than the 85^{th} percentile at both test sites.

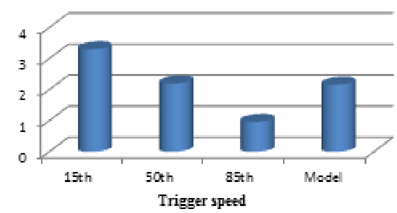


Figure 3. Energy consumption in Ah for the clustering model compared to the models based on the 15th , 50th and 85th percentiles at Korsgård test site

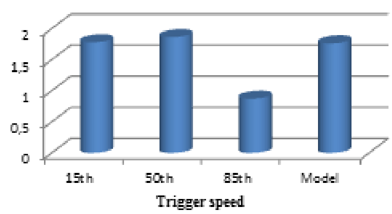


Figure 4. Energy consumption in Ah for the clustering model compared to the models based on the 15th , 50th and 85th percentiles at Mjälga test site

Fig. 5 and fig. 6 shows the effect of the clustering model on speed reduction compared to 15th, 50th and 85th percentiles. The speed reduction was based on the speed variation 100 meters before and 100 meters after triggering the sign. As seen in the figure, over the cause of one day, data were entered into three clusters according to the clustering algorithm.

 ISBN: 978-1-61208-352-0

The clustering model performed better at the Mjälga test site than at the Korsgård test site. At Korsgård, the greatest reduction in speed was in cluster one (nighttime) that had a trigger speed in the 15th percentile. At Mjälga, the greatest reduction that occurred at night was with a trigger speed that was in the 85th percentile.

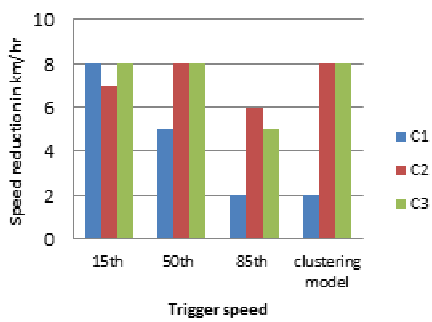


Figure 5. Speed reduction for the clustering model compared to the models based on 15th, 50th and 85th percentiles at korsgård test site

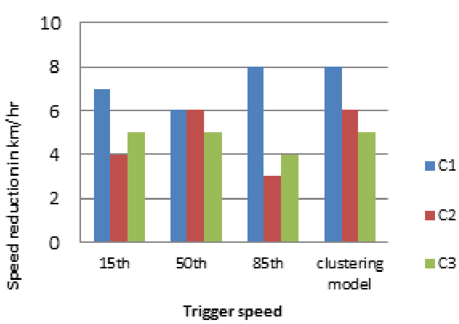


Figure 6. Speed reduction for the clustering model compared to the models based on 15th, 50th and 85th percentiles at Mjälga test site

In fact, the results obtained from both test sites were not similar where both sites have the same speed limit (40 km/hr). Note that. VAS should be individually configured and adapted to the location and its traffic conditions in order to result in optimal effectiveness.

VII. CONCLUSION AND FUTURE RESEARCH

This paper presents an intelligent algorithm that would help optimise the trigger point at which the sign should be operative. A self-organizing map helps the algorithm find the number of cluster and extract the right input dataset that is required to perform K-means clustering. K-means clustering divides the 24 hours of the day into 3 clusters. K-means clustering provides the algorithm to determine the trigger speed threshold simply by extracting the centroid median of the clusters. The comparison in energy consumption and speed reduction between the algorithm and other models shows that the algorithm is able maintain a reasonable level of energy consumption while also positively affecting driver behaviour. The algorithm was applied to the data set using traffic flow, median speed and standard deviation as the main features to categorize the time of day. To better assess the usefulness of this algorithm, the data set can be extended with more features. The data were collected by varying the trigger speed threshold value at each site. This data can be also extended by altering both the trigger speed threshold and the trigger distance threshold. Thus, it is beneficial to configure the sign in a way that reflects the actual traffic situation and provides optimal energy consumption.

Future research should also include a comparison between the effectiveness of the VAS during the day and at night as well as a comparison of the effectiveness of the VAS during the week and on the weekend. Meanwhile, the traffic data shall further be integrated with weather data sources. The results differ between the test sites. This means that the location of the sign can have an impact on the study. Obviously in order to optimise the effect of the trigger speed on a driver's speed, the VAS should be further adapted to a greater diversity of traffic and road conditions. In the long term the goal is to develop an "intelligent" VAS which self-adapts to traffic conditions on site in order to automatically operate at the optimum trigger speed.

REFERENCES

[1] S. Nygårdhs and G. Helmers, 'VMS - Variable Message Signs. A Literature Review', Swedish National Road and Transport Research Institue VTI rapport 570A, Linköping, 2007.

[2] S. Nygårdhs, 'Literature Review on Variable Message Sign 2006- 2009', Swedish National Road and Transport Research Institue VTI rapport 15A, Linköping, 2011.

[3] D. Jomaa, S. Yella, and M. Dougherty, 'Review of the effectiveness of vehicle activated signs', Journal of Transportation Technologies, Vol.3 No.2, 2013.

[4] Department for transport, 'Vehicle-Activated Signs', Traffic advisory unit, London, 2003.

[5] M. A. Winnett and A. H. Wheeler, 'Vehicle Activated Signs – a Large Scale Evaluation' TRL Report TRL 548, TRL Limited, UK, 2002.

[6] J. Mattox, W. Sarasua, J. Ogle, R. Eckenrode, and A. Dunning, 'Development and Evaluation of a Speed-Activated Sign to Reduce Speeds in Work Zones', Transportation Research, Record of the National Academies, Washington, 2007, pp. 3-11.

[7] L. K. Walter and J. Knowles, 'Effectiveness of Speed Indicator Devices on reducing vehicle speeds in London', Transport Research Laboratory TRL 314, Transport for London, London Road Safety Unit, 2008.

[8] U. A. Kumar and Y. Dhamija, 'Comparative Analysis of SOM Neural Network with K-means Clustering Algorithm', 2010 IEEE International Conference on Management of Innovation and Technology (ICMIT), Singapore, June 2010, pp .55-59.

[9] W. Huai-bin, Y. Hong-liang, X. Zhi-jian, and Y. Zheng, 'A Clustering Algorithm Use SOM and K-Means in Intrusion

Detection', 2010 International Conference on E-Business and E-Government (ICEE), Guangzhou, May 2010, pp. 1281 - 1284.

[10] X. Wang, W. Cottrell, and S. Mu, 'Using K-Means Clustering to Identify Time-of-Day Break Points for Traffic Signal Timing Plans', Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems, Vienna, Austria, September 2005, pp. 13-16.

[11] J. Vesanto, J. and E. Alhoniemi, E., 'Clustering of the Self-Organizing Map', IEEE Transactions on Neural Networks, vol. 11, no. 3, May 2000, pp. 586-600.

[12] Y. Chen, J. Hu, Y. Zhang, and X. Li, 'Traffic Data Analysis Using Kernel PCA and Self-Organizing Map', Proceedings of the 2006 IEEE on Intelligent Vehicles Symposium, Tokyo, Japan, June. 2006, pp. 472 – 477.

[13] M. J. Watts and S. P. Worner, 'Estimating the risk of insect species invasion: Kohonen self-organising maps versus k-means clustering', Ecological Modelling, Vol. 220, Issue 6, 2009, pp. 821–829.

[14] S. K. Shukla, S. Rungta, S., and L. K. Sharma, 'Self-Organizing Map based Clustering Approach for Trajectory Data', International Journal of Computer Trends and Technology, Vol. 3, No.3, 2012, pp. 321-324.

[15] T. V. Mathew, 'Fundamental Relations of Traffic Flow. Traffic Engineering and Management', IIT Bombay, 2012.

[16] G. J. Kerns, 'Introduction to Probability and Statistics Using R', First Edition, 2011.

[17] Hauser, T.A. & Scherer, W.T., 'Data Mining Tools for Real-Time Traffic Signal Decision Support and Maintenance', Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Tucson, AZ, Vol. 3, 2001, pp. 1471-1477.

Analytical Network Process for the On-line Dispatching

Boris Davydov
Institute of Natural Science
Far Eastern State Transport University
Khabarovsk, Russian Federation
e-mail: dbi@rambler.ru

Vsevolod Kuzmitskiy
Institute of Natural Science
Far Eastern State Transport University
Khabarovsk, Russian Federation
e-mail: vsevolod.kuzmitskiy@mail.ru

Andrey Serenko
Institute of Natural Science
Far Eastern State Transport University
Khabarovsk, Russian Federation
e-mail: a.serenko@mail.ru

***Abstract*-Analytical network process (ANP) is usually used for solving the complex multifactor problems when search for the rational decisions is made in stationary operating conditions with a large resource of time. In this paper, we propose the idea of development the ANP as applied to operational dispatching. The developed method is based on reducing the amount of information that is introduced into the decision support system via man-machine dialogue in the real time. Much of the data on the state of infrastructure and its adjustments comes from neighboring information systems. These data are used to specify the weights of the links between the elements presented in the ANP-model of the object. This algorithm simplifies the process of filling the estimated supermatrix and saves a lot of time. Also, simplified variants of interface are proposed to serve the operative data entering through the dialogue of operator with the DSS. The paper provides an example of applying the modified ANP in solving the problem of railway traffic dispatching.**

Keywords-Intelligent transport system; real-time management; railway dispatching; decision making; analytical network process (ANP)

I. INTRODUCTION

Decision making to overcome the complicated nonstandard situation is interrelated with processing a large amount of information. The expert analysis of an object or a process in its development is usually carried out over rather a long period of time. A lot of documentary sources and opinions of various experts are referred to, when analyzing the object. There are a number of effective methods to solve the specified problems, which are based on the use of special evaluating scales or the pairwise comparisons. The Analytical Hierarchy Process (AHP) and the Analytical Network Process (ANP) are the most often used expert methods [1].

In many operational cases there is a problem of solving a difficult situation when little time is allocated to find a rational solution. Many problems of operational management of such large systems as railways or power system relate to the tasks of this kind. Nonstandard situations regularly arise in these systems and there are no ready-made algorithms to find a way out. In these cases a large effect is obtained by using methods of expert assessments mentioned above. The expert analysis methods can be effectively applied in solving the railway traffic dispatching problems. Their properties make it possible to obtain a compromise solution when there is a conflict among managers of the control center [2].

Section 2 of this paper provides an overview of the well-known publications on the problem of expert analysis, which allows solving complex multifactor problems. Section 3 considers the emerging problems of railway dispatching. Section 4 offers a modified ANP method in application to the task of operative dispatching. It also presents the idea of forming such an interface which would significantly simplify the process of estimating the factors in the phase of information input into the Decision Support System (DSS). Section 5 sets out the conclusions and directions for further research.

II. OVERVIEW OF THE RELATED LITERATURE

At present there are a number of methods in decision theory, which make it possible to analyze the multifactorial problems. Based on these methods, the algorithms produce decomposition of the object into its elements through the pairwise comparisons; next, a calculation of the matrix which includes the weights of these elements. Most of the works that develop these expert methods consider only the algorithm for calculating the priority vector [3]. Other works improve the technique for handling the input (primary) data while retaining the scaling mechanism [4]. Most of the publications are devoted to the use of classical techniques AHP / ANP in solving practical problems [5][6]. These techniques are also implemented in the decision support systems: Technique of Order Preference by Similarity to Ideal Solution (TOPSIS), Cognitive Hierarchy Process (CHP), Fuzzy Cognitive Hierarchy Process (FCHP) [7]-[9], and others. We are not aware of familiar with any studies, which would formulate and solve the problem of sharing algorithms that improve the data entry and processing in the expert determination of the optimal solutions.

Many researchers work at the real-time technical conflict resolution problem in a railway [10]. Most of them use determinate modeling of the railway traffic. This approach allows accurate predicting the future evolution of the traffic on the basis of the actual train positions and speeds, as well as the signaling and safety system constraints. Assessing the possibility of using the algorithm of expert evaluations in

 ISBN: 978-1-61208-352-0

the process of object management in real time, Kabir et al. [4] have the opinion that it is impossible to solve the multifactor problem in a few short steps, as this will result in unacceptable inaccuracy. In our present paper we attempt to overcome this difficulty by means of using a priori information which comes through auxiliary channels. Besides, we try to create a rational construction of exchanging data between the operator and the DSS. Our goal is to create the DSS that allows dispatcher to get the intellectual and quick help in difficult cases of the railway traffic control, i.e., to obtain the solution of multifactor problem as soon as possible (within a few minutes).

III. ANP-MODEL FOR THE RAILWAY DISPATCHING

Selection of the decision on trains flow management is a typical problem of the railway dispatching. This is often a difficult task since we must take the priorities of the trains, the wagons, the work time of locomotive crews and a lot of other factors into account. Such difficult tasks are the typical multifactor scheduling problems.

Technical, commercial and organizational factors are taken into account in the decision-making process for the rational traffic management of trains. In addition, issues of providing the train traffic safety are constantly in sight of the manager. There are deviations from the normal operation that regularly occur on the railroad. These deviations often have a unique character because every time their reason is a peculiar combination of different factors. In many cases, simulation model is a poor description of the work of the railway section, as the relevant information about the process may be absent or tardy. In this case, operating decisions are made by the dispatcher. Intelligent DSS gives the opportunity to build an integrated dynamic model of the process by combining the actual data, which come on-line from the rail section, and the information that is entered by the operator. It is possible to use an efficient algorithm to determine the adjustments' priorities vector, based on the expert ANP-analysis.

ANP-model contains a number of technical clusters that correspond to the work of the stations, the train locomotives and the service divisions: power, signaling, communications and others. The separate cluster reflects the commercial weights of trains. The chief dispatcher of a railway line estimates the size of arcs which are directed to elements of this segment. Duty managers, who control the sets of locomotives, cars and other equipment, determine the intensity of the links between other specialized elements of the model. The common work of managers and the DSS is made in a dialogue mode that gives synergistic effect when determining the management decisions.

IV. MODIFIED ANP-MODEL AND RATIONAL MAN-MACHINE INTERFACE

ANP-model is created by decomposition of the goals, the function conditions of the system and the results of the process execution. The elements of a decomposed problem are grouped into the clusters, each of which reflects one of the components of the large structural elements of the object under analysis. The elements are connected by the links, and movement along the links leads, ultimately, to the final alternative solutions (see Figure 1). Each dispatcher assesses the elements which relate to his duties.

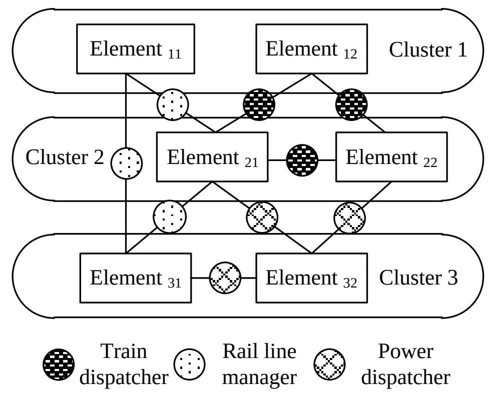


Figure 1. Fragment of network model

The corresponding weight of each of the final decision is formed on the basis of transition matrix, which is called the supermatrix in the ANP-approach. Controller of the certain object (we call him the rail line manager) introduces the values of DSS components, i.e., intensity of the links, which reflect the assessment of the importance of each of the factors. Herewith, he is using a certain estimating scale, mostly - from 1 to 9.

The process of scaling becomes very consuming in the case of multifactor model and the presence of cross-links between the elements. The expert makes his judgments and enters the relevant data into the DSS by using a technique of pairwise comparisons. When solving the multifactor problem, this process becomes very time-consuming and laborious. Similar problems can go quickly in the work of the manager, one after another, which leads to a significant increase of the facts and paired comparisons together with psychological strain. In the present research we propose to reduce the number of operations that are performed by the dispatcher in the estimation process in on-line mode. These reductions are made in the following way.

At the first stage, the dispatcher selects the type and location of the scenario which is considered in determining a specific decision, e.g., appointing the locality and time of the implementation of the adjustment and its type. This selection is reflected in the fact that the process model is simplified. Simplification is due to the fact that some of the elements are excluded by the dispatcher from the model as well as their respective links. These excluded elements have little influence on the result of the adjustment in this particular situation.

The second stage is represented by two sub-stages. In one of them part of the data, relating to commercial priorities, is entered by the rail line manager, who is responsible for this range of issues. Information that reflects the state of the infrastructure is introduced by the dispatchers of the relevant services: locomotives, power supply and others (fatty links). Considering the fragment of the model it is evident that the number of links (dotted links) that require estimation by the dispatcher is significantly reduced (see Figure 2).

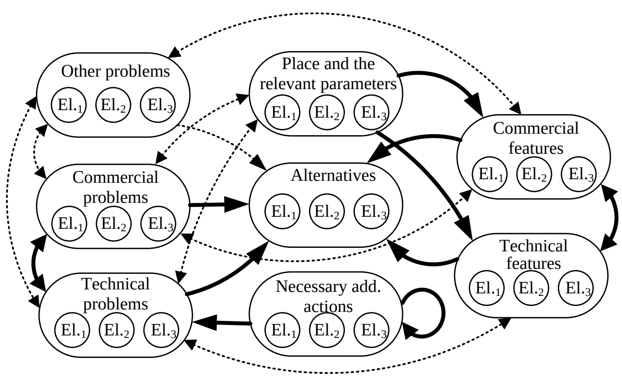


Figure 2. ANP network with some a priori and a posteriori cluster links

The volume of work that is performed in this second sub-step, is significantly reduced. This will ensure the achievement of the goal, namely, relieving the rail line manager from the large volume of non profile work. This allows him to focus on finding the most rational solution, which leads to conflict prevention or mitigation of its consequences.

The numerical estimation of intensity in each of the factors is used in the classical scheme of expert analysis. Furthermore, the operator has difficulty in establishing the consistency between different acts of pairwise comparisons. In the on-line method, which is proposed in this paper, non-standard ways of presenting the operator’s judgments are used.

One way assumes that operator selects a point in the area which is bounded by the axes of intensity relationships between elements using an interactive pen (see Figure 3). This example shows the pairwise comparison of elements from the estimation position based by two criteria. Intensity axes are the criteria for comparing the elements (e.g., technical and economic criteria). Curved lines show the mutually beneficial nature of the interaction of the criteria by which the operator compares elements.

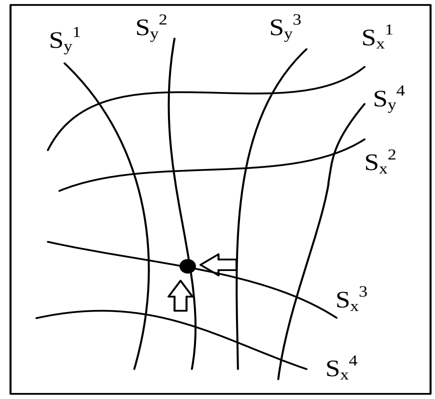


Figure 3. Existence domain of particular solution

The intensity value of the corresponding connection between a pair of elements is plotted on each axis. When the rail line manager compares the elements, he marks the point with these coordinates, which corresponds to his intuitive understanding of the features intensity. The second way is that a bar graph should be displayed on the screen for the use by the manager. A specific zone in the diagram is given for the each link (Figure 4).

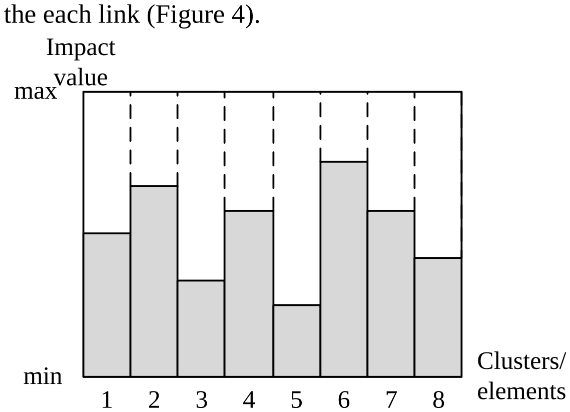


Figure 4. Mechanism of interlocking ratings

When making judgments on the intensity of features, the dispatcher makes mark in the certain areas corresponding to each connection between elements. This enables the manager to clearly see the ratio of their intensities, which speeds up the process of the data entering. Using these methods allows a person who makes the judgments, to connect the image thinking and, thus, simplify and accelerate the dialogue with the DSS.

V. Conclusion and Future Research

The article describes the ANP-model which is designed for the determination of rational solutions in terms of lack of time. The decision making process is accelerated due to obtaining estimates from adjacent experts and their use in a single supermatrix. Furthermore, the volume of the

 ISBN: 978-1-61208-352-0

processed information is reduced due to the fact that the dispatcher indicates the type of scenario and other information that reduces the dimensionality of the model. The paper also offers a convenient interface that simplifies the process of making dispatcher judgments.

In the future, we are planning to show how to solve the particular problems of railroad dispatching using the new ANP model. Development of the ideas in this work is expected in the direction of collective decision-making in conditions that require a compromise.

ACKNOWLEDGMENTS

Many thanks to Mr. Alexander Korchugov, the head of department of the Far Eastern Railway Dispatching Center for his great help in carrying out the case studies. Thanks are due to referees for their constructive comments. This work was supported by Russian Foundation for Basic Research, under research project № 12-08-98513-p_vostok_a.

REFERENCES

[1] T. L. Saaty and L. Vargas, Decision making with the analytic network process. Pitsburg, PA: Springer, 2006.

[2] B. Davydov, “Collective Intelligent Management of Freight Trains' Flow,” The Second International Conference on Intelligent Systems and Applications (INTELLI 2013), Apr. 2011, pp. 147-152

[3] M. Karel and M. Gavalec, “On-line fuzzy optimization based on cognitive hierarchy process,” 9th International. Conference Fuzzy Sets Theory and its Applications (FSTA 08), Feb. 2008, pp. 174-195

[4] M. A. Kabir, H. H. Latif and S. Sarker, “A multi-criteria decision-making model to increase productivity: AHP and fuzzy AHP approach,” Int. J. Intelligent Systems Technologies and Applications, vol. 12, 2013, pp. 207-229.

[5] M. Karel and M. Gavalec, “Multi-criteria models in autonomous decision making systems,” 10th International Symposium on the Analytic Hierarchy Process (ISAHP 2009), Aug 2009, pp. 257-263

[6] G. Longo, E. Padoano, P. Rosato and S. Strami, “Considerations on the application of AHP/ANP methodologies to decisions concerning a railway infrastructure,” 10th International Symposium on the Analytic Hierarchy Process (ISAHP 2009), Aug 2009, pp. 334-348

[7] H. H. Chen and H. Gu, “A fuzzy ANP model integrated with Benefits, Opportunities,Costs, and Risks to prioritize intelligent power grid systems,” Mathematical Problems in Engineering, vol. 34, Dec. 2013, pp. 259-269

[8] C. M. Wu, C. L. Hsieh and K. L. Chang, “A hybrid multiple criteria decision making model for supplier selection,” Mathematical Problems in Engineering, vol. 34, Dec. 2013, pp. 425-434

[9] C. F. Camerer, T. H. Ho and J. K. Chong, “A cognitive hierarchy model of games,” The Quarterly Journal of Economics, vol. 12, Aug. 2004, pp. 861-898

[10] J. Törnquist, “Computer-based decision support for railway traffic scheduling and dispatching: A review of models and algorithms” Proceedings of algorithmic approaches for transportation modeling, optimization and systems (ATMOS 2005), Oct. 2005, pp. 232-246

 ISBN: 978-1-61208-352-0

Constructing Autonomous Multi-Robot System

Nikola Šerbedžija
Fraunhofer FOKUS
Berlin, Germany
Nikola.Serbedzija@fokus.fraunhofer.de

Abstract—Developing control systems for swarm robotics require advanced techniques that can ensure adaptive, autonomous, self-aware and intelligent behavior. An engineering response to such demands is an ensemble based approach that structures a complex control system into dynamic ensembles of relatively simple system elements, called service components. The dynamism and autonomous behavior of the system elements are modeled by the knowledge- and predicate-based communication principle that allows for late (at run-time) evaluation of communication and connection rules among the system elements. The approach is illustrated on a concrete multi-robot scenario.

Keywords--swarm robotic; autonomous systems; development life cycle, ensemble-based system.

I. MOTIVATION

Constructing a multi-robot system requires multidisciplinary approach that calls for advanced techniques from the domains of software engineering, parallel and distributed system, agent systems and artificial intelligence. Each of the target disciplines poses grand challenges in its own field [1][2]. To respond to changing demands over a long operational time, adaptive and autonomous behavior at both individual and collective level [3] as well as energy awareness [4][5] need to be ensured.

The solution offered here responds to all these challenges. The approach decomposes a complex system into high number of service components – functionally simple building blocks enriched with knowledge attributes [6]. The knowledge of a component controls autonomic behavior at a local level. To ensure meaningful grouping and autonomy at higher levels (collective autonomy), system components are grouped into ensembles according to predicates over the components' attributes (which represents the major novelty of the approach). These predicates are actually implicit rules for communication bindings and represent global knowledge of the ensemble.

In order to guarantee correct and timely behavior in such demanding circumstances, this approach relies on formal methods. The system design and development phases are strictly defined leading to step-wise process of modelling, development, verification and validation.

The emphasis of this paper is on major engineering phases of the ensemble development lifecycle. A strongly pragmatic approach is illustrated by the concrete multi-robot scenario.

The paper is structured into six sections describing motivation (section one), engineering approach (section two), problem description (section three), system modelling using the SCEL language and JRESP framework (section four) and the deployment (section five). The conclusion (section 6) summarizes the achievements and indicates further directions for the work to come.

II. ENGINEERING APPROACH

Autonomous systems introduce a number of requests which are not present in other less dynamic systems. Constant changes both in the controlled environment and in the system *per se* require an appropriate methodology. The development process needs to be continuous, allowing for re-consideration and refinement both during the system development and during the system execution time. The approach described here proposes a persistent process for ensemble construction that consists of two major development circles, each having three phases:

- Design circle consists of:
  1. Requirement analysis,
  2. Modelling and programming, and
  3. Validation and verification phases.
- Runtime circle contains of
  1. Monitoring,
  2. Awareness, and
  3. Self-adaptation phases.

Two transitions, namely deployment and feedback ensure the correlation among the two circles.

- Deployment is a step-wise transition that is the result of modeling and programming phases. It begins with the first release and later continues whenever system modification occurs (re-deployment).
- Feedback is a transition that represents re-engineering, i.e., a system modification caused by problems discovered within the monitoring, awareness or self-adaptation run-time phases.

To ensure rigorous development of complex distributed autonomous systems, a number of tools and methods have been developed to support each of the phases and transitions within the development life cycle [7]. This paper focuses on tools and methods for modelling and the deployment, namely the SCEL (Service Component Ensemble Language) [8], and the jRESP (Java Runtime Environment for SCEL Programs) [9] and ARGoS[10] frameworks.

III. PROBLEM SPECIFICATION

Swarm robotics deals with creation of multi-robot systems that through interaction among participating robots and their environment can accomplish a common goal, which would be impossible to achieve by a single robot. To illustrate the

 ISBN: 978-1-61208-352-0

application from the swarm robotics domain a search and rescue scenario is presented.

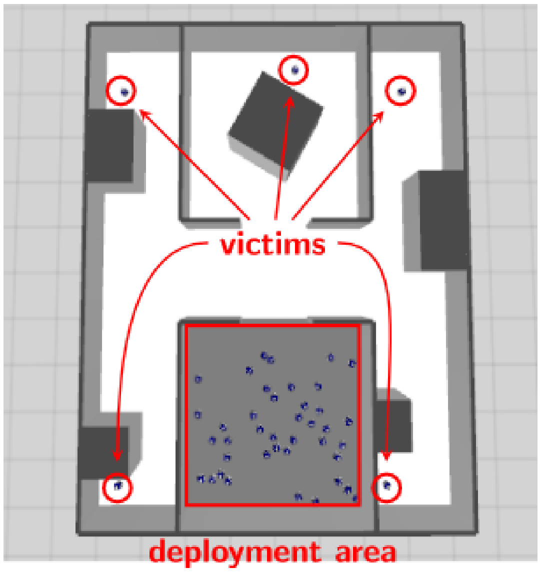


Figure 1. Scenario

A. *Swarm Robotics Scenario*

The basic idea behind the scenario is to organize and control a rescue operation in an emergency situation. Figure 1 illustrates the scenario where "an explo-sion happens in a nuclear plant causing the radiation, spill and collapse of a part of the building where a number of victims is trapped. To prevent further harm to human lives, a team of robots is deployed in the endangered area. The robots must explore the area, search for victims, and coordinate to save the victims as fast as possible. Besides removing victims, robots have to neutralize the radiation source by building blocks around it".

In the above scenario, a swarm of robots is distributed in a so called deployment area. The robots must reach the zone according to the scenario goal (finding victims and radiation source, carrying blocks, etc). Robots are not informed about the position of the targets. To discover their location they perform random walk combined with coordinated exploration. As soon as a robot reaches a radiation zone or a victim, it 'publishes' its location within the local knowledge repository. In this way, robots with the same task can be informed about the location of the corresponding target. Informed robots can then move directly towards the target thus saving time and energy.

Robots possess limited battery lifetime. To behave in an energy-aware manner, the robots must monitor the battery charge over the course of the experiment. If the battery charge drops too low, self-healing actions are required, e.g., reaching a charging station or sending a distress signal.

There are two types of robots in a multi-robot system needed to solve the "search and rescue" problem, as specified in the given scenario (see Figure 1): a - foraging robots that explore the environment and find objects and b - robots with a gripper, which can carry objects.

a) Foraging robot b) Robot with a gripper

Figure 2, Swarm robots

B. *Generic System Properties*

To further explore the control system requirements, the given scenario is closely examined and the major system characteristics are extracted (formulated in a generic form in order to keep them applicable in other application scenarios):

1. Individual goals
2. Coordination and distribution
3. Sharing and collectiveness (global goals)
4. Awareness and knowledge
5. Energy awareness and optimization

Each robot from the swarm has an individual goal (ie. simple task it can do). To solve a collective assignment, robots dynamically gather in a swarm, which further requires coordinated and distributed behavior. Knowledge of own capabilities and conditions as well as of those from the environment, bring awareness at both local and global level. Throughout its operational time, each robot from a swarm needs to observe its battery state and to adapt its functioning appropriately.

In a summary, a typical swarm robotics control system is highly collective, constructed of numerous independent entities that share common goals. Its elements are both autonomous and cooperative featuring a high level of self-awareness, self-expressiveness.

C. *Specific Scenario Properties*

In order to accomplish the rescue mission from the given scenario the robots need to perform the following operations: (i) efficient operation as robot energy depletes, and (ii) reaching consensus on the order in which the victims must be saved.

 ISBN: 978-1-61208-352-0

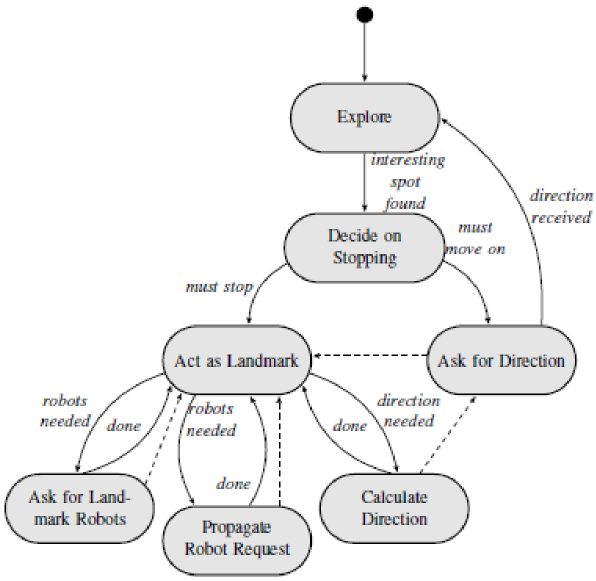


Figure 3. Robot behaviour graph

To solve the rescue scenario, a specific robot behavior called distributed exploration, is further specified. Robots are divided in two groups: workers and landmarks. Workers are robots that perform the actual rescuing task, transporting the victims to the deployment area. Landmark robots explore and mark important locations in the environment. Landmark robots are deployed first. They exit the deployment area one by one, moving straightforward until they encounter either a branching or an important location (e.g., a victim), or they are about to lose connectivity with the previous robots. Landmark robots form a network that is used by next approaching landmark robots.

Figure 3 shows a behavior graph of the "landmark" robot from a swarm. Based on the graph, the robot behavior is further specified, modeled, simulated and finally deployed on real robots.

IV. MODELING AND PROGRAMMING

Valid modeling and programming techniques ensure later correct behavior. The ensemble development lifecycle (EDLC) [7] uses a rigorous modeling/programming approach that allows for both formal reasoning on system properties and semi-automatic programming and validation.

A control system is decomposed into simpler hierarchical elements [9] called service components (SC) - representing simple functional entities with clearly defined individual goals, and service component ensembles (SCE) - representing a collection of service components with clearly defined collective goals.

Both components and ensembles have local knowledge used to express their goals. Knowledge is represented in terms of system properties and the goals are attributes over these properties.

A. Modeling Language SCEL

The basic entity of SCEL - Software Component Ensemble Language is the notion of autonomic component $I[K; \Pi; P]$ that consists of the following elements:

- An interface I given in a form of attributes – visible to other components.
- Knowledge repository K containing information about component interface, requirements, major state attributes etc. Managing such knowledge allows for self-aware behavior and dynamic interlinking with other system components.
- A set of policies Π that manage the internal and external interaction.
- A set of processes P defines component functionality specific to both the application and the internal management of knowledge, polices and communication.

For specification of processes, SCEL features a process algebra, which is extended by knowledge manipulation actions: **get** – taking a knowledge field out of the knowledge repository (blocks if not present), **qry** – getting a value of knowledge field while keeping the field in the knowledge repository (blocks if not present), **put** – inserting a knowledge field into the knowledge repository. The knowledge manipulation actions may use direct addressing (including a special target **self**) as well as addressing using a predicate, in which case, the action is performed on the knowledge of all components that matches the predicate (implicitly, creating an ensemble). A fully detailed presentation of SCEL syntax and semantics can be found in [7][8].

B. Modeling the Robot Scenario in SCEL

Qualitatively, the behavior of a single robot could be modeled with the following SCEL fragment, where each component $I[K; \Pi; (AM[ME])\ ]$ has the following description:

$$ME \triangleq \mathbf{qry}(\text{"}controlStep\text{"}, ?X)@\mathsf{self}.(\mathbf{get}(\text{"}termination\text{"})@\mathsf{self}.ME)[X]$$

$$AM \triangleq P_{batteryMonitor}[P_{dataSeeker}[P_{targetSeeker}]]$$

The processes composing the autonomic manager AM are as follows:

$$\begin{aligned} P_{batteryMonitor} \triangleq\ & \mathbf{qry}(\text{"}batteryLevel\text{"}, \text{"}low\text{"})@\mathsf{self}.\\ & \mathbf{get}(\text{"}lowBattery\text{"}, \mathsf{false})@\mathsf{self}.\\ & \mathbf{put}(\text{"}lowBattery\text{"}, \mathsf{true})@\mathsf{self}.\\ & \mathbf{qry}(\text{"}batteryLevel\text{"}, \text{"}high\text{"})@\mathsf{self}.\\ & \mathbf{get}(\text{"}lowBattery\text{"}, \mathsf{true})@\mathsf{self}.\\ & \mathbf{put}(\text{"}lowBattery\text{"}, \mathsf{false})@\mathsf{self}.P_{batteryMonitor} \end{aligned}$$

$$\begin{aligned} P_{dataSeeker} \triangleq\ & \mathbf{qry}(\text{"}targetLocation\text{"}, ?x, ?y)@(\mathcal{I}.task = \text{"}task_i\text{"}).\\ & \mathbf{put}(\text{"}targetLocation\text{"}, x, y)@\mathsf{self}.\\ & \mathbf{get}(\text{"}informed\text{"}, \mathsf{false})@\mathsf{self}.\\ & \mathbf{put}(\text{"}informed\text{"}, \mathsf{true})@\mathsf{self} \end{aligned}$$

Furthermore, a foraging robot (*TargetSeaker*) is described as:

 ISBN: 978-1-61208-352-0

```
P_targetSeeker ≜ qry("lowBattery", ?low)@self.
    if (low) then {
      get("controlStep", ?X)@self.
      put("controlStep", P_lowBattery)@self.
      qry("lowBattery", false)@self. P_targetSeeker
    } else {
      qry("target", ?found)@self.
      if (found) then {
        get("controlStep", ?X)@self.
        put("controlStep", P_found)@self. P_doTask
      } else {
        qry("informed", ?informed)@self.
        if (informed) then {
          get("controlStep", ?X)@self.
          put("controlStep", P_informed)@self. P_targetSeeker
        } else {
          get("controlStep", ?X)@self.
          put("controlStep", P_randomWalk)@self. P_targetSeeker
        }
      }
    }
```

The autonomic behavior of each robot is realized by means of an autonomic manager (AM) controlling the execution of a managed element (ME). The autonomic manager monitors in a self-aware fashion the state of charge of a robot's battery and verifies whether the target area has been reached or not. Self-adaptation can be naturally expressed in SCEL by exploiting its higher-order features, namely the capability to store/retrieve (the code of) processes in/from the knowledge repositories and to dynamically trigger execution of new processes. The autonomic manager can replace the control step code from the knowledge repository, thus implementing the adaptation logic and changing the managed element's behavior. For example, when a robot becomes informed, it self-adapts (i.e., self-configures) through its autonomic manager in order to move directly towards the target area.

C. *Simulation and Validation in jRESP*

The jRESP [9] framework is a runtime environment that provides Java programmers with ability to develop autonomic and adaptive systems based on the SCEL concepts. SCEL identifies the linguistic constructs for modeling the control of computation, the interaction among possibly heterogeneous components, and the architecture of systems and ensembles. jRESP provides an API that permits using the SCEL paradigm in Java programs.

The architecture of a generic jRESP node is shown in Figure 4. Each node is executed over a virtual machine or a physical device that provides the access to input/output devices and to network connections. Each node aggregates a knowledge repository, a set of running processes/threads, and a set of policies. Structural and behavioral information about a node can be collected into an interface via a set of attribute collectors. Nodes interact through ports supporting both point-to-point and group-oriented communications.

The robot scenario modeled in SCEL (as described in the previous section) is programmed in jRESP in the following way. The process ME (managed element) is

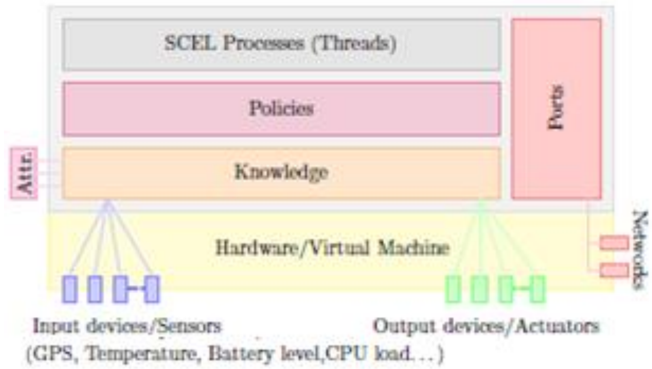


Figure 4. JRESP Architecture

rendered as an agent that continuously executes the control steps retrieved from the local knowledge repository:

```
public class ManagedElement extends Agent {
  public ManagedElement() {
    super("ManagedElement");
  }
  protected void doRun() throws Exception {
    while (true) {
      Tuple t = query(
        new Template(
              new ActualTemplateField("controlStep") ,
              new FormalTemplateField(Agent.class)) ,
          Self.SELF );
      Agent X = t.getElementAt(Agent.class, 1);
      X.call ();
    }
  }
}
```

The autonomic manager is modeled by the following three classes that provide a Java implementation for processes *P-batteryManager* and *P-dataSeeker* and *P-targetSeeker*, respectively:

```
public class BatteryMonitor extends Agent {
  public BatteryMonitor() {
    super("BatteryMonitor");
  }
  protected void doRun() throws IOException, InterruptedException{
      while (true) {
        query( new Template(
                  new ActualTemplateField("batteryLevel") ,
                  new ActualTemplateField("low")) ,
              Self.SELF );
        get( new Template(
                  new ActualTemplateField("lowBattery") ,
                  new ActualTemplateField(false)) ,
              Self.SELF );
        put( new Tuple( "lowBattery" , true ) , Self.SELF );
        query( new Template(
                  new ActualTemplateField("batteryLevel") ,
                  new ActualTemplateField("high")) ,
              Self.SELF );
        get( new Template(
                  new ActualTemplateField("lowBattery") ,
                  new ActualTemplateField(true)) ,
              Self.SELF );
        put( new Tuple( "lowBattery" , false ) , Self.SELF );
      }
    }

}
```

```
public class DataSeeker extends Agent {
  public DataSeeker() {
    super("DataSeeker");
  }
  protected void doRun() throws IOException, InterruptedException{
    Tuple t = query(new Template(
                          new ActualTemplateField("targetLocation") ,
                          new FormalTemplateField(Double.class),
                          new FormalTemplateField(Double.class)),
                       new Group(new HasValue( "task" , 1 ) ));
    double x = t.getElementAt(Double.class, 1);
    double y = t.getElementAt(Double.class, 2);
    put( new Tuple( "targetLocation" , x , y ) , Self.SELF );
    get( new Template(
              new ActualTemplateField("informed") ,
              new ActualTemplateField(false)) , Self.SELF );
    put( new Tuple( "informed" , true ) , Self.SELF );
  }
}

public class TargetSeeker extends Agent {
  public TargetSeeker() {
    super("DataSeeker");
  }
  protected void doRun() throws IOException, InterruptedException{
    while (true) {
```

A screen dump of a jRESP simulation of the robotic scenario is shown on Figure 5a, illustrating the movements of the foraging robots with a landmark searching algorithm.

Formal modelling of the multi-robot scenario also contributes to the validation phase of the software development lifecycle. As shown on figure 5b, the jRESP simulation can be used to calculate the probability of finding victim in the given scenario (for the given algorithm, the probability of success is directly proportional to the number of landmarks used in a search). The verification of the search algorithm is shown on the figure 5c, insuring that the algorithm will always converge. The simulation, validation and verification tools all refer to the problem described in the scenario shown on the Figure 1.

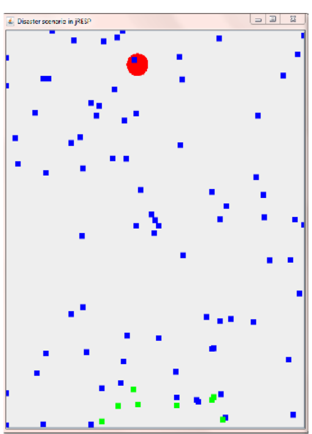
V. DEPLOYMENT

The deployment transition of the ensemble development life cycle involves the implementation of the robot behaviors on real robots. This step is the most critical in robotics because it is usually the most expensive, time-consuming, and risky. For this reason, deployment is usually performed in two distinct phases. The first phase consists of testing the robot behaviors in accurate physics-based simulations. These simulations must include as many details as possible, so as to minimize costly issues in the next phase. The next deployment phase consists of testing the behaviors on the real platform with robots.

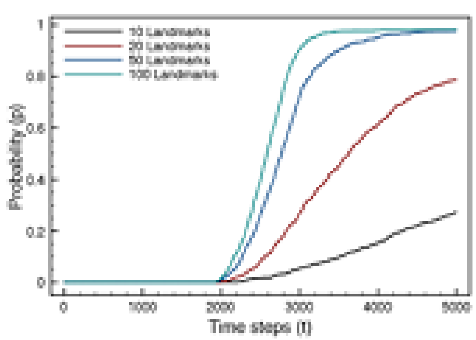
For the deployment purposes the ARGoS (discrete-time simulator for multi-robot systems) [11] platform is used as it provides both an efficient simulation framework and a straightforward deployment with real robots. The same control system is firstly tested on a simulated environment and then is transferred to the real platform, substituting simulated robots with the real ones..

ARGoS is a physics-based multi-robot simulator. It aims to simulate complex experiments involving large swarms of robots of different types in the shortest possible time. It is designed around two main and often contradictory requirements: efficiency - achieving high performance with large swarms, and flexibility - allowing the user to customize the simulator for specific experiments.

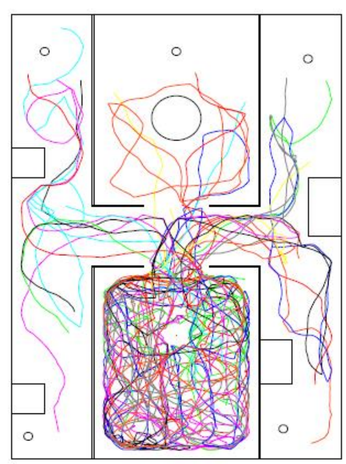
To bridge the efficiency and flexibility gap, ARGoS system deploys a number of novel design choices. First, in ARGoS, it is possible to partition the simulated space into multiple sub-spaces, managed by different physics engines running in parallel. Second, ARGoS' architecture is multi-threaded, thus designed to optimize the usage of modern multi-core CPUs. Finally, the architecture of ARGoS is highly modular. It is designed to allow the user to easily add



a) Simulation with foraging and the landmark robots



b) System validation



c) System verification

Figure 5. Screen dumps from the simulation, validation and verification tools

custom features (enhancing flexibility) and to allocate computational resources where needed (thus decreasing run-time and enhancing efficiency).

The final deployment phase in a real robot setting is still being developed. In preparation for the final deployment, simultaneously with ARGoS simulation, the two types of robots have been further refined (Figure 2).

VI. CONCLUSION

This paper presents an integrated approach to model, validate and deploy ensemble-based multi-robot systems. The non-centralized character of the approach allows for autonomic and self-aware behavior, which is achieved by introduction of knowledge elements and enrichment of compositional and communication primitives with awareness of both system requirements and individual state of the computing entities.

The essence of the ensemble-based approach is to de-compose a complex system into a number of generic components, and then compose the system into ensembles of service components. The inherent complexity of such ensembles is a huge challenge for developers. Thus, the whole system is decomposed into well-understood building blocks, reducing the innumerable interactions between low-level components to a manageable number of interactions between these building blocks. The result is a so-called hierarchical ensemble, built from service components, simpler ensembles and knowledge units connected via a highly dynamic infrastructure. Ensembles exhibit four main characteristics: adaptation, self-awareness, knowledge and emergence, providing a sound methodology for engineering autonomous systems. A number of analyses, modeling, programming and validation tools are under development and evaluation in different application settings [7].

The pragmatic significance of the approach has been illustrated by the multi-robot scenario showing the major design and development phases on the concrete practical example. The SCEL language [8] and jRESP [9] are used for modeling, programming and validating the scenario. Finally, ARGOS system [10] is used to fine-tune and deploy the control system in a real robot setting.

Further work is oriented towards monitoring and testing of the real system as well as towards analyses of the run-time behavior. These activities belong to the second cycle of the EDLC [7] and will be the subject of future work. Tools to monitor ensemble based systems should be developed that allow for run-time analyses and verification of awareness and self-adaptive behavior of both system elements and the system as a whole.

ACKNOWLEDGMENTS

Most of the work presented here has been done under the ASCENS project (project number FP7- 257414) [7] funded by the European Commission within the 7th Framework Programme. Special thanks go to Roco de Nicola (CNR), for the work on SCEL [8], Michele Loreti (University of Florence) for the work on jRESP [9], and Carlo. Pinciroli (Brussels University) for the work on ARGoS[10].

REFERENCES

[1] I. Sommerville et al. Large-scale complex it systems. Commun. ACM. 2012, Vol.55, No.7, pp.71-77

[2] M. Hoelzl, A. Rauschmayer, and M. Wirsing. Engineering of software-intensive systems. In Software-Intensive Systems and New Computing Paradigms, LNCS, 2008, Vol.5380, pp.45-63.

[3] E. Bonabeau, M. Dorigo, and G. Theraulaz. Swarm Intelligence: From Natural to Artificial Systems, New York, NY: Oxford University Press, Santa Fe Institute Studies in the Sciences of Complexity, 1999, ISBN 0-19-513159-2.

[4] B. Degener, B. Kempkes, and F. Meyer. Energy-Awareness in Self-organising Robotic Exploration Teams .Organic Computing, Springer, 2011, pp346-365.

[5] C. Seo. Energy-Awareness in Distributed Java-Based Software Systems. In Proc. of the 21st IEEE International Conference on Automated Software Engineering (ASE'06). IEEE, 2006, pp.343-348.

[6] M. Hoelzl et al. Engineering Ensembles: A White Paper of the ASCENS Project. ASCENS Deliverable JD1.1. http://www.ascens-ist.eu/whitepapers [retrieved: May 2014].

[7] Project ASCENS (Autonomic Service-Component Ensembles). http://www.ascens-ist.eu ASCENS [retrieved: May 2014].

[8] R. De Nicola, M. Loreti, R. Pugliese, and F. Tiezzi. SCEL: a language for autonomic computing. Technical Report. Universita a degli Studi di Firenze. Available at: http://rap.dsi.unifi.it/scel/ [retrieved: May 2014].

[9] M. Loreti, jRESP: a run-time environment for scel programs. Technical Report. Universita a degli Studi di Firenze. Available at: http://rap.dsi.unifi.it/scel/, [retrieved: May 2014].

[10] C. Pinciroli et al. ARGoS: a Modular, Parallel, Multi-Engine Simulator for Multi-Robot Systems. Swarm Intelligence, Springer, Berlin, Germany, 2012, vol. 6, no. 4, pp 271-295.

 ISBN: 978-1-61208-352-0

Fuzzy Logic Control for Gaze-Guided Personal Assistance Robots

Carl A. Nelson
Department of Mechanical and Materials Engineering
University of Nebraska-Lincoln
Lincoln, NE, USA

***Abstract*—As longer lifespans become the norm and modern healthcare allows individuals to live more functional lives despite physical disabilities, there is an increasing need for personal assistance robots. One of the barriers to this shift in healthcare technology is the ability of the human operator to communicate his/her intent to the robot. In this paper, a method of interpreting eye gaze data using fuzzy logic for robot control is presented. Simulation results indicate that the fuzzy logic controller can successfully infer operator intent, modulate speed and direction accordingly, and avoid obstacles in a target following task relevant to personal assistance robots.**

Keywords—gaze tracking; fuzzy logic; autonomous robot; obstacle avoidance; personal assistance robot

I. INTRODUCTION

With lifespans increasing worldwide due to advancements in healthcare and related technologies, the importance of care for the elderly and disabled is increasing. In particular, there is a shifting emphasis in technology development towards improving quality of life in the face of diminishing physical capabilities. One of the burgeoning areas of this trend is personal assistance robotics. In a typical scenario, a robot assistant may be present in the home to help with basic day-to-day tasks (e.g., object retrieval), especially those tasks requiring navigation throughout the home, since age- or disability-related mobility limitations may keep an individual from performing all these tasks personally. In extreme circumstances, it can even be challenging to give instructions to the robotic assistant, as in the case where the individual is not physically able to type, speak, or otherwise provide clear inputs to the human-robot interface. Here, we present preliminary progress designing a robotic assistance system which relies on gaze tracking, including eye blinking patterns, to infer a person's intent and thereby create instructions for the robot. In this paper, we specifically focus on the intelligent inference of intent based on gaze and blinking input.

This problem is an extension of the task of robotic target following and path planning. Significant work has been done in this area of service robotics, where a robot is to follow a moving target. For instance, some have used computer vision, using optical flow algorithms to track the target [1][2]. Other computer vision-based approaches have used Kalman filters for improving the accuracy of tracking [3]. Other tracking methods include the use of depth images with verification via a state vector machine [4], or following acoustic stimuli [5]. Control approaches in these target-following scenarios include potential field mapping [6] and a variety of other techniques. Of particular interest are fuzzy logic controllers [5][7][8], which tend to be used primarily for steering. Here, we will describe a fuzzy logic controller which not only determines the robot's heading based on the location of the target, but also avoids obstacles and modulates speed based on the perception of intent from the combined gaze direction and blink frequency inputs. This is conceptually based in part on recent work demonstrating how such a combined input using operator gaze could be used for automatic control of endoscope positioning in surgical tasks [9] using a commercially available eye tracking system, which is also similar to the work described in [10]. This approach extends beyond the most typical uses of eye gaze, which tend to be for two-dimensional human-computer interfaces [11].

The remainder of this paper is organized as follows. In section II, the eye gaze data and the fuzzy logic controller are described. In section III, simulation results are presented. Section IV includes conclusions and recommendations for future work.

II. METHODS

A. Test Dataset and Simulation

A gaze dataset was artificially generated to have spatiotemporal characteristics similar to those described in [9], in a planar workspace. The data were arbitrarily assumed to be sampled at 10 Hz and included a logical *blink* data channel in addition to the x and y gaze target channels on the interval [-0.5 0.5], providing a total of over 23 seconds of simulated robot tracking. Due to the noisy nature of gaze data, the target X was determined by a linear weighted average of the previous n data points P, with n = 20:

$$X_k = \frac{2}{n}\sum_{i=k-n}^{k}\left(1-\frac{k-i}{n}\right)P_i. \tag{1}$$

In this particular dataset, there are five intended target locations, characterized by dwelling gaze and higher blink frequency, and it is assumed that a supplementary action such as object placement or retrieval would follow target acquisition (although this supplementary action is beyond the scope of this preliminary study). Within the workspace, three round obstacles were defined to test the ability of the simulated robot to avoid obstacles while seeking a target. The data were imported into MATLAB (The MathWorks, Natick, MA) for simulation of gaze-based robotic target tracking.

B. Fuzzy Logic Controller

A Mamdani-type fuzzy logic controller [12] with five inputs and three outputs was created using the Fuzzy Logic Toolbox in MATLAB; the Mamdani-type model handles multi-input, multi-output problems better than the Sugeno-type alternative. The inputs, shown in Table I, were intended to take into account the distance to the target, the degree of uncertainty of the target's position, and the presence of obstacles in the path from the robot's position to the target. The outputs, also shown in Table I, were used to control the speed and heading of the robot, including steering adjustments for obstacle avoidance. All of the membership functions were triangular, as shown in Fig. 1.

TABLE I. FUZZY CONTROLLER VARIABLES AND THEIR TRIANGLUAR MEMBERSHIP FUNCTIONS EXPRESSED IN MODAL FORM [LOWER BOUND, MODE, UPPER BOUND]

Input/ Output	Variables		
	Name	***Units***	***Membership Functions***
I	Target Δx	distance	negative [-1, -0.5, 0] zero [-0.1, 0, 0.1] positive [0, 0.5, 1]
I	Target Δy	distance	negative [-1, -0.5, 0] zero [-0.1, 0, 0.1] positive [0, 0.5, 1]
I	Target variability	distance	zero [-0.1, 0, 0.1] low [0.05, 0.25, 0.45] high [0.35, 1, 1.4]
I	Blink frequency (normalized)	-	zero [-0.4, 0, 0.4] low [0.1, 0.5, 0.9] high [0.6, 1, 1.4]
I	Obstacle distance	distance	zero [-0.2, 0, 0.2] low [0, 0.3, 0.6] high [0.35, 1, 1.4]
O	Speed	distance/ time	zero [-0.4, 0, 0.4] low [0.1, 0.5, 0.9] high [0.6, 1, 1.4]
O	Heading	rad	up [0.125, 0.25, 0.375] up/right [0, 0.125, 0.25] right [-0.125, 0, 0.125] down/right [0.75, 0.875, 1] down [0.625, 0.75, 0.875] down/left [0.5, 0.625, 0.75] left [0.375, 0.5, 0.625] up/left [0.25, 0.375, 0.5]
O	Heading adjustment	rad	zero [-0.4, 0, 0.4] low [0.1, 0.5, 0.9] high [0.6, 1, 1.4]

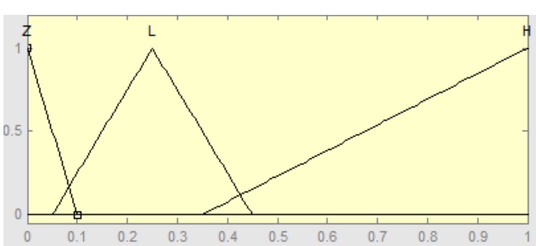


Fig. 1. Membership functions for target variability (zero, low, and high).

The target was determined using a weighted average of the gaze data as in (1), the target and obstacle distance variables were then calculated using the Pythagorean theorem, and target variability was represented by the standard deviation of the gaze input data over the averaging window. Blink frequency was normalized to the interval [0 1] by assuming that four blink events within the 20-sample averaging window was high (achieving a value of 1), and lower blinking rates in the same window of time receive a proportionally smaller membership value. If no obstacles were detected in the direct path between the robot and target, the obstacle distance was set to its maximum value of 1.

Concerning the output variables, the maximum speed was constrained to a value of 0.25 (covering one-fourth the workspace in one second at maximum speed), and the maximum heading adjustment for obstacle avoidance was set at ±100°. The heading variable was scaled to allow the robot to steer within the full 360° range.

Fifteen rules were defined to characterize the influence of the five input variables on the three outputs. In particular, four rules capture the influence of the inputs on the output variable *speed*, eight rules accommodate the division of heading into eight regions in polar coordinates, and the remaining three rules govern obstacle avoidance. The rules defining the fuzzy logic controller are as follows:

1. IF *blink* IS *high* THEN *speed* IS *high*
2. IF *target Δx* IS *positive* OR *target Δx* IS *negative* OR *target Δy* IS *positive* OR *target Δy* IS *negative* THEN *speed* IS *high*
3. IF *target Δx* IS *zero* AND *target Δy* IS *zero* THEN *speed* IS *zero*
4. IF *target variability* IS *high* OR *blink* IS *low* THEN *speed* IS *low*
5. IF *target Δx* IS *positive* AND *target Δy* IS *zero* THEN *heading* IS *right*
6. IF *target Δx* IS *positive* AND *target Δy* IS *positive* THEN *heading* IS *up/right*
7. IF *target Δx* IS *positive* AND *target Δy* IS *negative* THEN *heading* IS *down/right*
8. IF *target Δx* IS *negative* AND *target Δy* IS *zero* THEN *heading* IS *left*
9. IF *target Δx* IS *negative* AND *target Δy* IS *positive* THEN *heading* IS *up/left*
10. IF *target Δx* IS *negative* AND *target Δy* IS *negative* THEN *heading* IS *down/left*
11. IF *target Δx* IS *zero* AND *target Δy* IS *positive* THEN *heading* IS *up*
12. IF *target Δx* IS *zero* AND *target Δy* IS *negative* THEN *heading* IS *down*
13. IF *obstacle distance* IS *zero* THEN *heading adjustment* IS *high*

 ISBN: 978-1-61208-352-0

14. IF *obstacle distance* IS *low* THEN *heading adjustment* IS *low*

15. IF *obstacle distance* IS *high* THEN *heading adjustment* IS *zero*

The first four rules govern the robot's speed. Higher blink rates imply a more focused operator intent and cause increased speed (rule 1). Conversely, high gaze variability or low blink rate imply a less sure target and lead to lower speed (rule 4). The higher the distance to the target, the higher the necessary speed to reach it in a timely manner, and speed should drop to zero as the target is reached (rules 2-3). It should be noted that lower speeds are sometimes desirable to conserve energy either when the goal is unclear or has been reached.

Rules 5-12 pertain to heading. These are relatively straightforward and use the four cardinal directions and the four semi-cardinal directions to navigate in the planar map based on the relative target distance in the *x* and *y* directions.

The remaining three rules constitute the robot's obstacle avoidance behavior. The closer the obstacle, the larger the heading adjustment applied to go around it. Whether this adjustment is added or subtracted from the heading variable is determined by whether the obstacle centroid is to the right or the left of the straight line along the robot's heading.

III. Results

Simulation in MATLAB revealed the ability of the fuzzy logic controller to simultaneously determine human intent from the combined gaze location and blink data, use this intent to modulate robot speed, follow a moving target, and avoid obstacles. In Fig. 2, it can be observed that the robot (whose position is indicated by red diamond markers) can start at a location somewhat removed from the initial target, quickly acquire the target, and then follow it consistently without colliding with obstacles in the workspace.

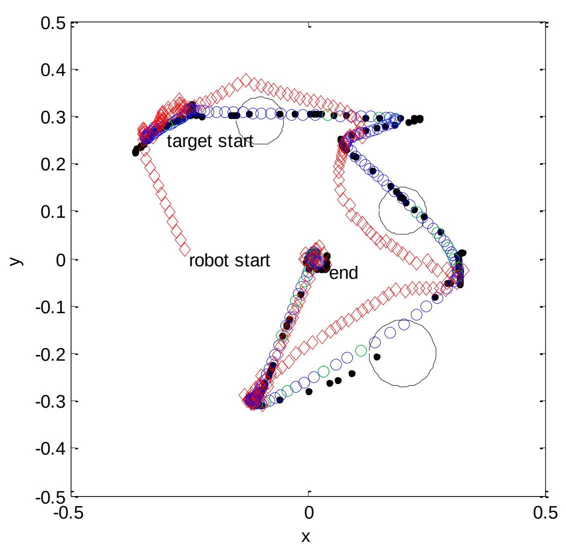


Fig. 2. Target following behavior: robot (red diamond markers) follows target (blue circles) while avoiding fixed environmental obstacles. Green circles indicate target location with a *blink* event. Targets of definite interest (based on dwell duration and blink frequency) are at approximately (-0.3, 0.3), (0.1, 0.3), (0.3, 0), (-0.1, -0.3), and (0, 0). Raw gaze data are shown as black dots.

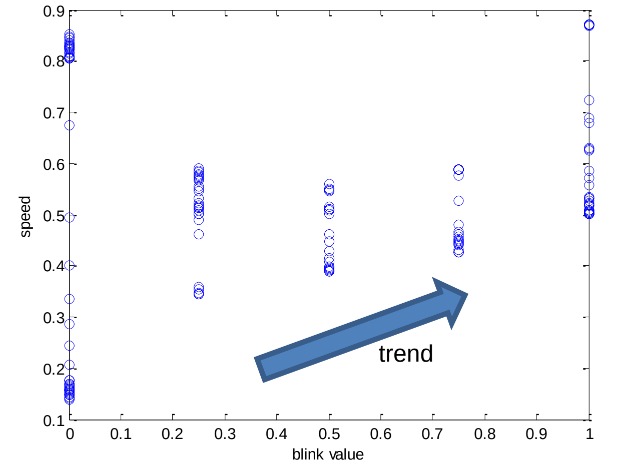


Fig. 3. Output speed as a function of *blink* membership function value: a positive correlation is noted.

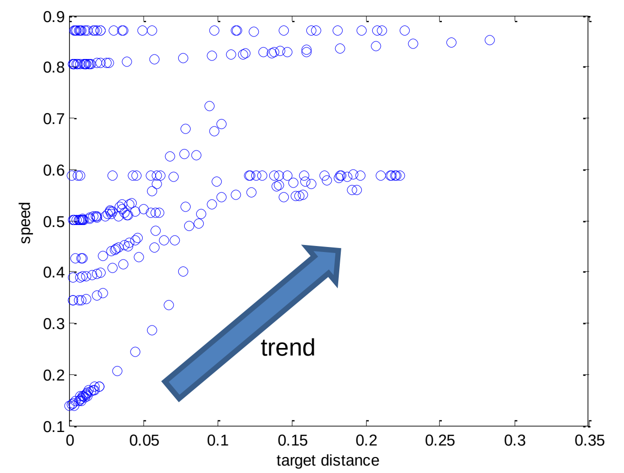


Fig. 4. Output speed as a function of target distance: a positive correlation is noted.

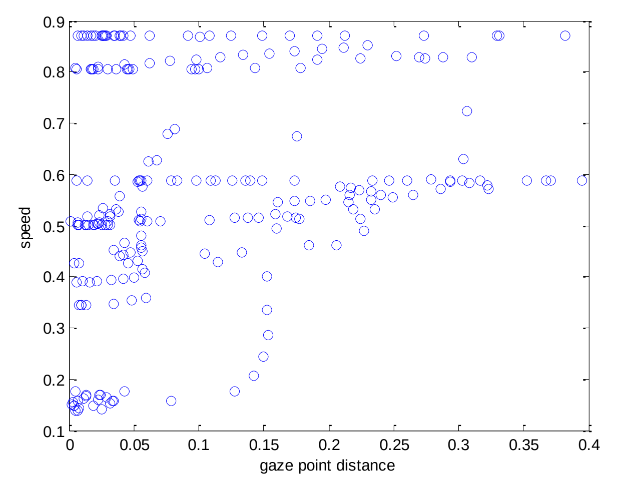


Fig. 5. Output speed as a function of distance to current gaze location: correlation is much less pronounced.

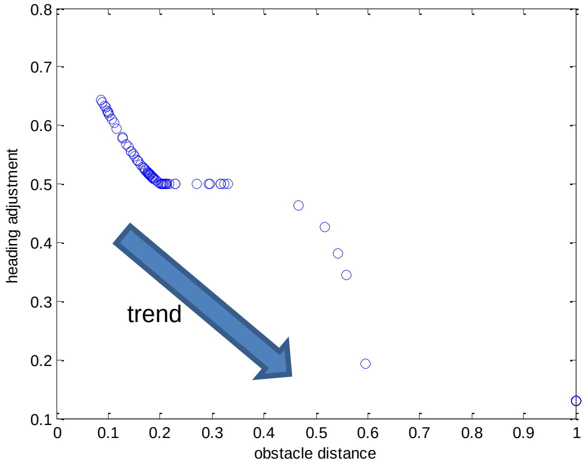


Fig. 6. Modulation of heading adjustment based on obstacle distance demonstrates effective obstacle avoidance.

The more interesting outcomes of the simulation are highlighted in Figs. 3-6. In Fig. 3, one can see that robot speed tends to increase with blink frequency, as intended. High speed at low blink can be attributed to the effects of target distance (particularly at the beginning of the simulation). Note that the results in Fig. 3 are striated at discrete levels, since blinking is a discrete, logical event; this could be smoothed by applying an averaging method similar to that used in target determination. Target distance also has an important effect on speed, as shown in Fig. 4. In contrast, Fig. 5 illustrates that the relationship between robot speed and distance from the robot to the actual gaze point is less pronounced, since the target is based on a weighted average of the gaze point and is thus a less noisy signal. The interdependence of speed on multiple input parameters is evident in Figs. 3 and 4. The effectiveness of the obstacle avoidance behavior is shown in Fig. 6 by the clean heading adjustment curve.

IV. CONCLUSIONS

In this paper, a new technique for gaze-based guidance of personal assistance robots has been illustrated. Fuzzy logic allows the robot to simultaneously manage multiple behaviors, practicing energy conservation when appropriate but pursuing the target when human intent to do so is clear. Combined use of the eye gaze point and blinking data is a pivotal feature of the fuzzy logic controller. Basic obstacle avoidance is demonstrated as an integrated behavior within this controller.

The preliminary results presented in this paper suggest promise for additional future work. The fuzzy controller should be tested using actual gaze data acquired from human users using an eye tracking system, with and without a real-time robot presence, to determine how visual feedback between the robot and human may affect human gaze input. The controller should also be tuned for improved performance, and some of its more basic rules may be replaced by a more sophisticated steering and obstacle avoidance rule set. More advanced work will focus on detailed implementation for a broader variety of personal assistance tasks (e.g., object pick-and-place, operating on a static object) in a true 3D environment.

ACKNOWLEDGMENT

Contributions by X. Zhang to the motivating concept of this paper, and support for this research from US NSF award 1264504, are gratefully acknowledged.

REFERENCES

[1] J. Woodfill, R. Zabih, and O. Khatib, 1994, "Real-time motion vision for robot control in unstructured environments," Proc. ASCE Robotics for Challenging Environments, pp. 10-18.

[2] P. K. Allen, A. Timcenko, B. Yoshimi, and P. Michelman, 1993, "Automated tracking and grasping of a moving object with a robotic hand-eye system," IEEE Transactions on Robotics and Automation, vol. 9, no. 2, pp. 152-165.

[3] T.-S. Jin, J.-M. Lee, and H. Hashimoto, 2006, "Position control of mobile robot for human-following in intelligent space with distributed sensors," International Journal of Control, Automation, and Systems, vol. 4, no. 2, pp. 204-216.

[4] J. Satake and J. Miura, 2009, "Robust stereo-based person detection and tracking for a person following robot," Proc. IEEE International Conference on Robotics and Automation 2009, Workshop on People Detection and Tracking, Kobe, Japan, May 2009.

[5] J. Han, S. Han, and J. Lee, 2013, "The tracking of a moving object by a mobile robot following the object's sound," J Intell Robot Syst, vol. 71, pp. 31–42.

[6] C.-H. Chen, C. Cheng, D. Page, A. Koschan, and M. Abidi, 2006, "A moving object tracked by a mobile robot with real-time obstacles avoidance capacity," Proc. of the 18th International Conference on Pattern Recognition (ICPR'06), 4 pp.

[7] M. Mucientes and J. Casillas, 2005, "Learning fuzzy robot controllers to follow a mobile object," International Conference on Machine Intelligence, Tozeur, Tunisia, Nov. 5-7, 2005, pp. 566-573.

[8] M. Abdellatif, 2013, "Color-based object tracking and following for mobile service robots," International Journal of Innovative Research in Science, Engineering and Technology, vol. 2, no. 11, pp. 5921-5928.

[9] X. Zhang, S. Li, J. Zhang, and H. Williams, 2013, "Gaze Contingent Control for a Robotic Laparoscope Holder," J. Med. Devices, vol. 7, no. 2, pp. 020915.1-020915.2.

[10] D. P. Noonan, G. P. Mylonas, A. Darzi, G.-Z. Yang, 2008, "Gaze Contingent Articulated Robot Control for Robot Assisted Minimally Invasive Surgery," IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, Sept. 22-26, 2008, pp. 1186-1191.

[11] A. Leonel, F. B. de Lima Neto, S. C. Oliveira, H. S. B. Filho, 2011, "An Intelligent Human-Machine Interface Based on Eye Tracking to Afford Written Communication of Locked-In Syndrome Patients," Learning and Nonlinear Models, vol. 9, pp. 249-255.

[12] P. Hájek, 1998, Metamathematics of Fuzzy Logic, Kluwer Academic Publishers, Dordrecht, The Netherlands.

Establishing a Lightweight Communicative Multiagent Java Framework

Braxton McCraw, Justin Ruger, Roberto A. Flores
Department of Physics, Computer Science & Engineering
Christopher Newport University
Newport News, USA
braxton.mccraw.09@cnu.edu
justin.ruger.07@cnu.edu
roberto.flores@cnu.edu

Robert C. Kremer
Department of Computer Science
University of Calgary
Calgary, Canada
kremer@cpsc.ucalgary.ca

***Abstract*—This paper presents work in progress towards building CASA$_{LITE}$, which is a Java programming framework to create lightweight, communicational, hybrid multiagent systems. Our goal is to create a framework that runs on small and large devices with a minimal footprint (lightweight), that relies in message communications as the basic mechanism of interaction (communicational), and that allows building a mix of agents ranging from purely software-based to robotic-based (hybrid). To validate our work we plan test cases for single robot control and multiple robot collaboration. CASA$_{LITE}$ will adopt a robot simulator for offline testing of robot programs.**

Keywords: multiagent; robotic; programming; framework.

I. Introduction

When considering the advancement of technology, it is important not only to consider the importance of solving new problems but also finding ways to improve upon existing solutions. Perhaps no field of study in computer science has contributed more to the automation and efficiency of optimizing problem solving than artificial intelligence. The traditional view in this discipline holds that with a more complex problem we must construct a more complex solution, mostly through more sophisticated abilities for an individual problem-solver component. An alternative approach is to use not one complex (and usually expensive) component but several low-cost units to handle separate parts of the problem. However, it has been observed that having large team sizes and a greater variety of components raises the complexity of the system [1].

On an orthogonal dimension to multiagency we find the means of implementation, where systems are not limited to purely software or purely hardware components: it is increasingly common to find hybrids (such as in robotics), where versatile software programs are imbedded in the control of complex hardware devices. The focus of our research lies in this area, where groups of multiple separate components (named agents) work together on a task (either cooperatively or additively) and together comprise what is referred to as a multi-agent system (MAS).

Agents can have varying degrees of cooperation and communication between them as well as a range of decision-making independence. Agents can either have the same nature and abilities (homogenous) favoring tasks that are scalable via agent addition, or have different specializations (heterogeneous), favoring applications that benefit from a division of labor. The greatest strength of a multiagent approach is the low coupling afforded by the modularity of its components. General multi-agent frameworks can then be tailored to an application's requirements, leading to a world of possible implementations [2].

Multi-agent systems, while providing a useful abstraction are not fully adopted yet for general use. This can be attributed to a lack of awareness of the potential of agents working in tandem, small publicity of successfully implementations, over-expectations of early adopters of agent technologies, aversion to taking risks on a relatively young and unproven technology, and the lack of developmental and design tools for creating agent systems has led to trepidation of investing time and money into widespread multi-agent applications [3].

Although MAS is an appealing abstraction to organize complex systems, we are concerned with the lack of appropriate tools to implement such systems and, in particular, hybrid communicative MAS. A hybrid framework would allow the implementation of potentially mixed populations of software-controlled hardware agents (e.g., sensors, robots) and purely software agents (e.g., centralized coordinators, decision-makers) that communicate through explicit messaging to organize and coordinate their actions. In our experience, several frameworks could be used to implement such systems; among them are Player/Stage, MS Robotics Developer Studio, JADE and CASA. Given the objectives defined by their creators, the features in these frameworks cannot squarely be compared vis-à-vis. However, these features to a varying degree make them amenable to hybrid MAS implementations.

In this paper, we present as our contribution our early efforts implementing CASA$_{LITE}$, a small-footprint framework to build hybrid communicational multiagent systems. Our framework is planned to be lightweight for deployment in a range of devices, from small and embedded devices (such as SUN/Oracle SPOT [4]) to hand-held devices (such as phones and tablets), and computers with larger capacities. We chose Java as the implementation language to maximize the array of devices in which we could deploy our framework and for its suitability as a familiar language for undergraduate students. As a test case, we'll deploy a CASA$_{LITE}$ agent on an Android tablet that both interfaces with an iRobot Create and is able to exchange messages and video feedback to a remote CASA$_{LITE}$ agent running in a laptop. To validate the appropriateness of CASA$_{LITE}$ as a multiplatform framework, we'll also implement a $\text{CASA}_{\text{LITE}}$ agent for the NAO humanoid robot [5]. To validate $\text{CASA}_{\text{LITE}}$ as a collaborative framework we will implement multi-robot collaboration to solve a maze. Lastly, $\text{CASA}_{\text{LITE}}$ supports inter-agent communication through socket-based streams transmitting text messages with LISP-like syntax compliant with KQML (Knowledge Query Manipulation Language) [6] and FIPA (Foundation for Intelligent Physical Agents) [7].

The remainder of the paper is organized as follows. Section 2 discusses MAS taxonomy and frameworks considered as existing alternatives to implement hybrid communicational MAS. Section 3 briefly describes an overview of the design of $\text{CASA}_{\text{LITE}}$, and Section 4 presents our planned experiments and conclusions.

II. MAS Taxonomy & Frameworks

In one dimension, agents are organized by their degree of sophistication, from simple reactive components to components of massive decision-making complexity. In another dimension, agents are organized by their degree of collaboration, from isolated components to components that can function in teams and organizations [8]. In one other dimension, agents are organized by their behaviors; for example, grazing – where a robot traverses an area [9]. Naturally, this task is enhanced with a multi-agent approach, since several robots can coordinate their actions to cover an area faster. In such cases, the coordination mechanism must survive the worst-case scenario of unit loss, which should not be handled on a unit-to-unit basis but rather as a collective [10] enabling the team to continue working even in cases of unit loss [11]. We assume that message communication is a coordinating mechanism that complements or even subsumes other coordination mechanisms afforded by the environment. For example, a robot waiting for a block to be moved by another agent could perceive that the block has been moved (thus making the block the coordinating device) or could wait until the agent pursuing the task notifies that the block has been moved (thus making the message the coordinating device). In the former case, it is assumed that agents have the awareness and comprehension to know when a block has been fully moved (c.f., acting when the block is in an intermediate and not final moving state) whereas the latter waits until the agent responsible for the moving action has cleared its completion. Our approach is to assume that messages are the intrinsic coordination device and that agents use communication to coordinate their actions.

Our initial approach towards finding a suitable hybrid MAS platform was to survey existing frameworks to identify candidates. Our ideal framework would be hybrid and multiplatform (able to implement robot interfaces and software agents), provide a simulator (in cases where hardware is not readily available) and communicative (it must support autonomous communication to enable explicit collaboration between agents). An additional requirement is that its communications comply to some degree to standards such as KQML and FIPA. The first framework identified was JADE (Java Agent Development Framework). Being written in a familiar language made JADE an attractive option as well as its inclusion of several Java packages that could be used as-is or modified for the specific platform. However, JADE lack of a robotic simulator yielded a less attractive option than other frameworks [12]. JADE has been used to implement MAS for human-robotic interaction [13], for coordinating a citywide taxi ordering service [14], and an intelligent hotel booking system [15] (further examples can be found in [16]).

We also explored Player [17], which is a robotic device server bundled with a robot simulator named Stage. Stage is relatively lightweight, and is able to simulate hundreds of robots on a standard desktop PC. Communications in Player are achieved through socket streams (which is in line with our goals), making it compatible with programs written in languages supporting sockets. In addition to the Stage simulator, the biggest draw to Player is its simplicity, since the server core has been simplified and reworked to the point where all the functionality is in a single thread of execution. Player's lightweight approach has been explored in large distributed systems, such as the DARPA SDR program, which implemented a 100-robot experiment [18]. On the other hand, there have been reports of compatibility issues between Player drivers and certain robot models, with some problems being operating system specific [19].

Robotics Developer Studio (MSRDS) [20][21] is Microsoft's development environment for designing robot applications across a variety of programming languages. MSRDS has support for iRobot Create and LEGO Mind storm platforms. MSRDS features a robust simulator that can be adjusted through user-made scripts to define simulation parameters. The simulator was the most enticing aspect of MSRDS, supporting simple user-defined polygons to represent the robot and obstacles in the environment. In our view it is the easiest to use framework investigated, although scripting was tedious and it does not lend itself well to modification.

The Collaborative Agent Systems Architecture (CASA) [22][23] is an elaborated framework for agent interaction written in Java. CASA has a robust message and conversational structure based on social commitments, and implements a basic robot simulator for iRobot Create. Its computational footprint, however, makes CASA an unlikely choice to implement agents for small devices.

After reviewing these frameworks, we weighted their communicational abilities, robotic simulation potential and programming fitness for undergraduate students and decided to explore redesigning the core functionality in CASA to support hybrid MAS systems.

III. $\text{CASA}_{\text{LITE}}$

$\text{CASA}_{\text{LITE}}$ is a small-footprint framework to build hybrid communicational multiagent systems. It distances itself from existing frameworks with its adaptability and simplicity while incorporating features from other implementations.

Figure 1 shows the core design of our framework. AbstractAgent is the super-class of all $\text{CASA}_{\text{LITE}}$ agents. It has an event hub (to queue and process events) and a message hub (to queue incoming and handle outgoing messages). Events (not shown) can be synchronous (agents wait for its completion) or asynchronous (executing

 ISBN: 978-1-61208-352-0

independently of an agent's thread), and can be recurrent (executing more than one time); if so, they can be timed to occur at intervals. Message hubs implement socket streams for receiving and sending text messages. Messages are text-base strings whose syntax is KQML/FIPA-compliant. An example message can be "(request :content (curve :speed 200 :radius -3000 :bump any) :language iRobot)". In the context of our iRobotCreate implementation, this message requests the robot to drive forward at a certain speed and radius (i.e., drive in a curve) and stop when the bump sensors detect an obstacle. Events can have an event handler reacting to the event's transitions, e.g., enqueue, dequeue, updates. Event handlers are useful to coordinate responses to message requests when the events created by these requests are resolved. AndroidAgent and NAOAgent (the latter not yet implemented) are agents that implement bare-bone scaffolding programs for their corresponding architecture (namely Android OS and NAO's OS, respectively). AndroidiRobotAgent is an agent interfacing with the core implementation of an iRobotCreate controller, which has its own event hub to queue Create specific commands. This abstract class is extended by either the hardware-aware class of a concrete Create instance (iRobotActual) or by the simulation compatible class (iRobotSimulated) that runs within a 2-dimensional simulator ported from CASA. As will be described in the next section we implement an Android agent as the robot controller running in a tablet sitting atop an iRobot Create robot. Communication between the tablet and the Create are supported through a Bluetooth connection. For practical purposes both the tablet and the Create are treated as one autonomous component.

IV. PLANNED EXPERIMENTS & CONCLUSIONS

We plan several test cases to assert the adequacy of our framework, first for robot control and then for collaboration.

Our first experiment will focus mostly on robot control, with minimal communicational interaction and decision-making. In particular, we will implement a $CASA_{LITE}$ agent in an Android tablet directly interfacing with an iRobot Create through a Bluetooth connection. As mentioned earlier, both the tablet and robot are considered a sole agent.

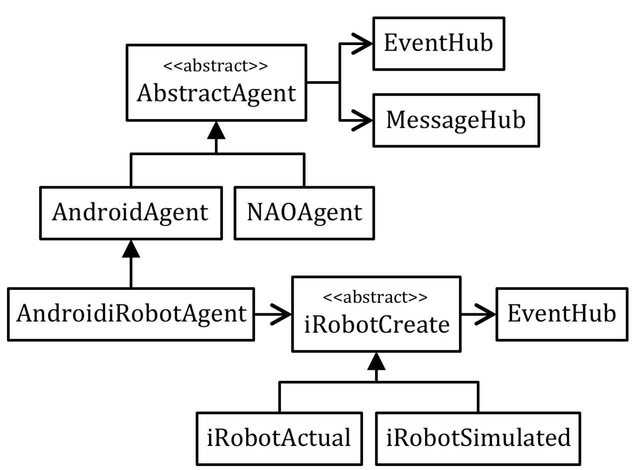


Figure 1. Overview of the main class hierarchy in $CASA_{LITE}$.

This agent will receive messages with commands to control the robot from another agent located in a remote laptop. Likewise, the laptop agent will receive notifications from the Android agent informing of the success or failure of submitted commands plus notifications about the state of the robot. We are designing these notifications under a subscription model, in which the laptop agent subscribes to state changes on the robot monitored by the Android agent, including changes to bumper, wall, floor and cliff sensors. In addition, we will program the Android agent to stream video to the remote laptop agent and to display text messages sent from the laptop agent. At the end, the iRobot/Android agent should be able to drive (guided by a human operator on the laptop agent) through our building and to a different floor by taking an elevator (by using its text interface to request the help of human bystanders to push elevator buttons) and return to the place where we deployed it.

Our second experiment will consist of multiple robots searching for an exit in a rectangular maze. Robots are deployed randomly without a priori knowledge of the environment although they will be aware of other robots through their communications. Initially robots are only aware of their immediate surroundings as afforded by their local sensors, and begin by traversing the maze in single-decision making mode, acquiring knowledge of the maze as they advance and using a simple search algorithm to identify paths to traverse. A different mindset takes over once agents come in contact with each other. At that point, agents communicate their individual maze mappings and combine them (taking as reference their point of contact) into common ground to start division of labor. Any new search paths are negotiated between these agents and any new map space discoveries are shared through their communications, with the potential to add other agents (either isolated or part of other clusters) as they come in contact with each other. Agents will continue exploring the maze until one of the robots finds an exit and communicates its location to all agents in its cluster. To facilitate traversal of the maze, robots will need an ultrasound sensor not currently provided on the Create. This feature will also need to be implemented in the robot simulator.

To conclude, in this paper we present our earlier efforts to build $CASA_{LITE}$, a Java-based framework for implementing lightweight, communicational, hybrid multiagent systems. Agents are programmed with basic communicational abilities to transmit KQML/FIPA-syntax compliant text messages through network streams as a way to enable collaboration. Currently, we have implemented the basic functionality of an abstract agent and the interface to the iRobot Create. Shortly we will begin implementing the Android agent to be deployed in our initial single robot test case scenario. After this test case we will integrate to $CASA_{LITE}$ the robot simulator from the CASA framework and use it to program the collaborative test case in which several robots communicate to find a maze exit.

 ISBN: 978-1-61208-352-0

V. Acknowledgements

The work presented in this paper is partially supported by the National Science Foundation under Grant Number (NSF 0841295), GK12 Program, CNU W.I.S.E.

References

[1] B. Gerkey and M.J. Matarić, "A formal analysis and taxonomy of task allocation in multi-robot systems." The International Journal of Robotics Research, volume 23, issue 9, pages 939-954, September 2004.

[2] K.P. Sycara, "Multiagent Systems."AI Magazine, American Association for Artificial Intelligence (AAAI), volume 19, number 2, pages 79-92, 1998.

[3] M. Pěchouček and V. Mařík, "Industrial deployment of multi-agent technologies: review and selected case studies." Autonomous Agents and Multi-Agent Systems, volume 17, issue 3, pages 397-431, 2008. doi: 10.1007/s10458-008-9050-0

[4] Sun/Oracle SPOT, Sun SPOT World [Online] Available from: http://www.sunspotworld.com/ [retrieved: May 2014]

[5] NAO, Aldebaran Robotics [Online] Available from: http://www.aldebaran-robotics.com/ [retrieved: May 2014]

[6] T. Finin, Y. Labrou, and J. Mayfield, "KQML as an agent communication language." In Software Agents, J.M. Bradshaw (Ed.), AAAI/MIT Press, pages 291-316, 1997.

[7] FIPA, Foundation of Intelligent Physical Agents [Online] Available from: http://fipa.org/ [retrieved: May 2014]

[8] A. Farinelli, L. Iocchi, and D. Nardi, "Multi-robot systems: A classication focused on coordination." IEEE Transactions on System Man and Cybernetics, Part B, pages 2015-2028, 2004.

[9] T. Balch and R.C. Arkin, "Communication in reactive multiagent robotic systems." Autonomous Robots, volume 1, issue 1, pages 27-52, 1994. doi: 10.1007/BF00735341

[10] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "A taxonomy for multi-agent robotics." Autonomous Robots, volume 3, issue 4, pages 375-397, 1996. doi:10.1007/BF00240651

[11] J. Ferber, "Multi-agent system: An introduction to distributed artificial intelligence." Addison Wesley Longman, 1999. ISBN 0-201-36048-9

[12] F. Bellifemine, A. Poggi, and G. Rimassa, "JADE–A FIPA-compliant agent framework." Proceedings of Practical Applications of Agents and Multi-Agents (PAAM), pages 97-108, London, 1999.

[13] V.S. Santos, C.P. Cândido, P. Santana, L.C. Correia, and J.B. Barata, "Developments on a system for human-robot teams." Conference on Autonomous Robot Systems & Competitions (Robótica 2007), volume 1, pages 1-7, Paderne, Portugal, 2007.

[14] A. Moreno, A. Valls, and A. Viejo, "Using JADE-LEAP to implement agents in mobile devices." EXP-In Search for Innovation (special issue on JADE), Telecom Italia Lab, volume 3, issue 3, 2003. [Online] Available from: http://jade.tilab.com/papers/EXP/02Moreno.pdf [retrieved: May 2014]

[15] C. McTavish and S. Sankaranarayanan, "Intelligent agent based hotel search & booking system." 9th WSEAS International Conference on Telecommunications and Informatics (TELE-INFO '10), pages 61-66, Catania, Sicily, Italy, May 29-31, 2010 [Online] Available from: http://wseas.us/e-library/conferences/2010/Catania/TELE-INFO/TELE-INFO-09.pdf [retrieved: May 2014]

[16] JADE, Java Agent Development Framework. [Online] Available from: http://jade.tilab.com/ [retrieved: May 2014]

[17] The Player Project: Software for robot & sensor applications. [Online] Available from: http://playerstage.sourceforge.net/ [retrieved: May 2014]

[18] M. Kranz, R.B. Rusu, A. Maldonado, M. Beetz, and A. Schmidt, "A player/stage system for context-aware intelligent environments." Proceedings of UbiSys'06, System Support for Ubiquitous Computing Workshop, 8th Annual Conference on Ubiquitous Computing (Ubicomp 2006), Orange County California, September 2006, pages 17-21. [Online] http://www.eislab.net/publications/2006/Rusu06UbiSys_preprint.pdf [retrieved: May 2014]

[19] D.S. Michal and L. Etzkorn, "A comparison of player/stage/gazebo and Microsoft robotics developer studio." Proceedings of the 49th Annual Southeast Regional Conference (ACM-SE '11), ACM, pages 60-66, Kennesaw, Georgia, 2011. doi: 10.1145/2016039.2016062

[20] K. Johns and T. Taylor, "Professional Microsoft robotics developer studio." Wiley Publishing, Indianapolis, 2008. ISBN 978-0-470-14107-6

[21] J.S. Cepeda, L. Chaimowicz, and R. Soto, "Exploring Microsoft robotics studio as a mechanism for service-oriented robotics." Proceedings of the 2010 Latin American Robotics Symposium and Intelligent Robotics Meeting (LARS '10), IEEE Computer Society, pages 7-12, 2010. doi 10.1109/LARS.2010.18

[22] R.C. Kremer, "CASA User Manual." [Online] Available from: http://casa.cpsc.ucalgary.ca/doc/CasaUserManual.pdf [retrieved: May 2014]

[23] R.C. Kremer, R.A. Flores, and C. La Fournie, "A performative type hierarchy and other considerations in the design of the CASA agent communication architecture." In F. Dignum (ed.), Advances in Agent Communication, Lecture Notes in Artificial Intelligence, Volume 2922, Springer-Verlag, pages 59-74, 2004.

Comparison of DEAS and GA for Sensitivity Optimization in MEMS Gyroscope

Hyunhak Cho
Department of Interdisciplinary Cooperative Course: Robot
Pusan National University, Busan, Korea
e-mail: darkruby1004@pusan.ac.kr

Moonho Park, Jaeyong Kim, Sungshin Kim
Department of Electrical and Computer Engineering
Pusan National University, Busan, Korea
e-mail: 82akakak, arioner, sskim@pusan.ac.kr

***Abstract*—This paper is research into how to optimize the sensitivity values of an installed gyroscope in the autonomous guided vehicle with a magnet-gyro guided system. A magnet-gyro guided system mostly uses a MEMS gyroscope, which is small-sized, uses little power, and costs little. However, the MEMS gyroscope needs a high sensitivity value for changing angular velocity in each environment, not only due to the necessity of an accurate sensitivity value for the measured angle value but also due to the difference between the measured angle and sensitivity value. The sensitivity value describes the specifications of the sensor, but the sensitivity value is influenced by the given environment or gradient. Therefore, the optimization process is required for the sensitivity value of the installed gyroscope in the environment. A number of optimization algorithms have been studied, but we chose the Dynamic Encoding Algorithm for Searches (DEAS) and the Genetic Algorithm (GA) to optimize the sensitivity value. We used an AGV with laser navigation for experiments in this paper. We did 5 experiments for each change of the rotation angle - 30, 40, 50, 60° - and compared the calculations of the sensitivity value of optimization through the DEAS and the GA. The experiment results confirm that the optimization sensitivity values of the DEAS contain less error than the optimization sensitivity value by the GA algorithm.**

Keywords - DEAS, GA, Gyroscope, AGV

I. INTRODUCTION

An Autonomous Guided Vehicle (AGV) is a large mobile robot that can load and unload freights to the appointed position within either a fixed or unfixed path. Because it does not require manpower, AGVs can reduce wages and fatal accidents. Localization and position estimations are examples that use AGV technology [1]-[3]. Localization is a categorized wire and wireless guide method. The wire-guided method is a means for an AGV with detection by an attached or laid guideline to the floor. This method is used for safety reasons, but requires installation and maintenance. The wireless guide method remedies the wire guide method's shortcomings. It induces an AGV through a laser or an attached landmark to the ceiling or surface of a wall instead of a floor guideline. The wireless guide method has the advantage of not requiring installation of a guideline by driving an imaginary guideline. A typical device that uses the wireless guide method is laser navigation. Laser navigation has a ±25*mm* localization accuracy. It is also easy to modify the driving line, and extend workspace. But this comes at the cost of a slow response speed of 250*ms*, a big error of localization at a high speed or rotation driving. The magnet-gyro guide method is a new method to gain the advantages from both the wire and wireless guide methods. This method can induce an AGV by the measured magnet field information through installed magnets in the floor. The measured angular velocity by a gyroscope when there is no magnet, when the magnet is detected, the position and the angle of an AGV revise by the information of the detected magnet [4][5]. This method does not require the trouble of installation and maintenance of the wire guided method, and resolves the high cost of the wireless-guided method. Accuracy of the gyroscope for a position estimation of the vehicle is of importance in the magnet-gyro guide method because it induces an AGV by the measured yaw angle by the gyroscope when driving the gap of magnets. Commercialized gyroscope with a high accuracy is restricted to the AGV, because of its big size and large power consumption. The gyroscope of a Micro Electro Mechanical System (MEMS) was used for AGV miniaturization and low power. However, the MEMS gyroscope has problems such as low bias safety, low accuracy through noises, and low performance of straightness in comparison to the commercialized gyroscope. Generally, to calculate the sensitivity value for the angular velocity of the MEMS gyroscope, the representative value of the sensor specification must be used. However, because the sensitivity value is influenced by the given environment or gradient, an optimization method is required for the sensitivity value of the installed gyroscope in the environment. The existing optimization methods that use differentiation are complicated and need many arithmetic operations. So calculating the optimized sensitivity value is very difficult. Dynamic Encoding Algorithm for Searches (DEAS) optimization algorithm fits a Micro Control Unit (MCU) with a limited performance because the DEAS algorithm searches a global optimization solution without differentiation. This paper makes a comparison between the DEAS and the Genetic Algorithm (GA) in order to make a performance evaluation of the DEAS optimization algorithm. This paper is organized as follows. Section 2 presents the measurement system for a MEMS gyroscope. Section 3 goes into details regarding DEAS and GA algorithms for the optimization sensitivity value of the MEMS gyroscope. Section 4 mentions the experiments conducted and the results. Lastly, Section 5 is the conclusion of this paper.

II. MEASUREMENT SYSTEM

A. Experimental system

To experiment using the proposed method, an axel-driven fork type AGV with a built-in laser navigation was used. Laser navigation can measure global location. It is installed on top of the AGV to protect disturbances, and measures global position by using reflectors.

 ISBN: 978-1-61208-352-0

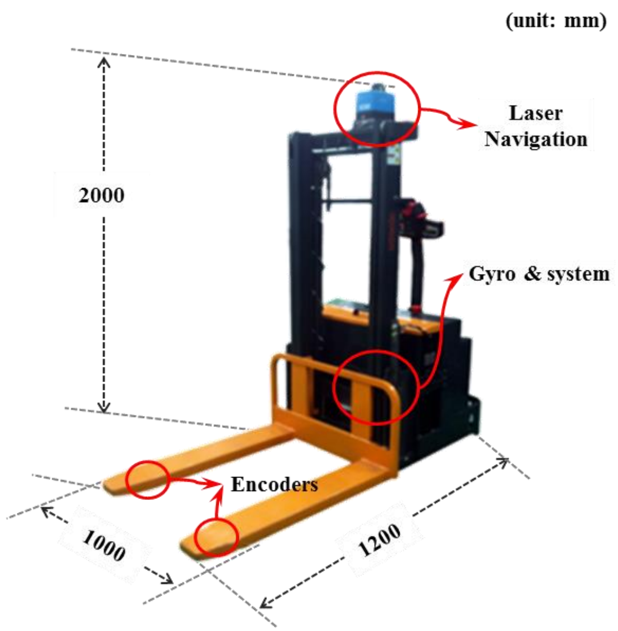


Figure 1. Forklift type AGV used in the experimet

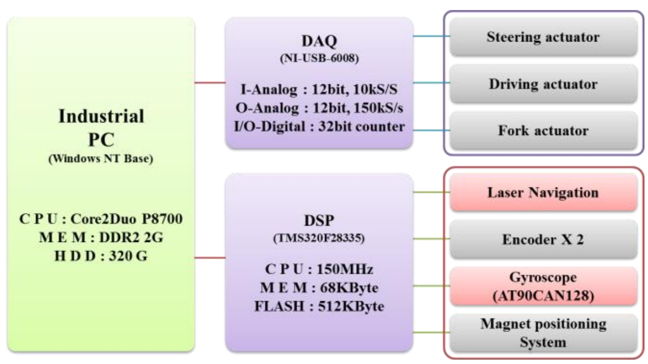


Figure 2. Hardware configuration of the AGV

The forklift-type AGV and the hardware configuration used in the experiment are shown in Figures 1 and 2, respectively. In Figure 2, industrial PC is used for rapid research and development. DAQ is used to control the driving parts of a manufactured forklift type. The position data of laser navigation is received through DSP every 500ms and the linear velocity of an encoder is received through DSP every 25ms. Transmission rates of MEMS gyroscope and magnet positioning system are 25ms and 100ms respectively. The data of the sensors is transferred from AVR to DSP. Encoders and gyroscope are used to measure a local position to compensate for the low response speed of laser navigation. The gathered data is sent to an industrial PC by DSP with RS-232 serial communication. Table 1 is the specification of installed sensors. The localization accuracy of laser navigation as a global positioning sensor depends on the number of recognized reflectors, and the distances between the sensor and reflectors. The sensor has a positioning accuracy of ±4mm and an angle accuracy of ±0.1°. MEMS gyroscope used in this experiment was designed by ADXRS613, AT90CAN-128 and 12bit ADC.

TABLE I. SPECIFICATION OF SENSORS

Sensor	Specification	
Laser Navigation (NAV200)	Supply voltage	24 V
	Positioning accuracy	±4 m～±25 mm
	Angular accuracy	±0.1 °
Gyroscope (ADXRS613)	Input voltage	5 V
	Range	±150 °/s
	Drift	±3 %
Magnet positioning system developed in our Lab.	Input voltage	5 V
	Sensitivity	10 mV/G
	Polarity	Bipolar (N/S)

B. Calculation of angular velocity using gyroscope

The output of MEMS gyroscope is influenced by temperature variation. The angular velocity value and the temperature value of the MEMS gyroscope are measured every 25ms. The output of gyroscope (O^+) depending on temperature is calculated using equation (1).

$$O^+ = O^- - ((2^{12}/2) - T) \times CA \qquad (1)$$

O^- and T is obtained through ADC data of each gyroscope. O^- is the raw data of the gyroscope and T is the temperature value. CA is the temperature constant and is set at 0.08. Equation (2) is the calculation method using the output value of a gyroscope.

$$\Delta\theta = (C - O^+) \times S \qquad (2)$$

C is the central output value of a gyroscope (0 ~ 4096: 2048). S is the sensitivity of the gyroscope, with an average value of 0.0125V/°/s in the datasheet. In this paper, the central value of the gyroscope is calculated by averaging 1000 data points obtained in the stop state of AGV. The ADC value of the gyroscope in a stop state is between 2034 and 2042. We used 2039 as the average output value. However, gyroscope has errors because the sensitivity of the installed environment of the sensor is not considered. To use MEMS gyroscope sensor as a navigation system, direct optimization is needed as a measurement system within a built-in MEMS gyroscope.

III. SENSITIVITY OPTIMIZATION

The sensitivity of gyroscope changes depends on the environment of the gyroscope's installed place and slope. To improve performance of the gyroscope, sensitivity optimization should be fulfilled in MCU with measured data in real time, because of the computation time and complex algorithm structure in existing optimization algorithms with stochastic or direct search, etc. Therefore the optimization algorithm for sensor needs a low computation for operating in real time. This paper proposes a simple and rapid method to optimize the sensitivity of the gyroscope through DEAS.

A. Dynamic encoding algorithm for searches

Existing optimization algorithms with derivative methods have high computation time. It is not suitable in real time.

 ISBN: 978-1-61208-352-0

Thus, sensitivity values should be directly optimized in MCU to consider the environment of a gyroscope. It is obvious that existing algorithms cannot be used if it has a high computation time. However, a DEAS algorithm can find both a local and global optimal solution of a cost function by using well-planned computer operations, which has a good operation time to performance [6][7]. The implementation of DEAS is simple. DEAS can also rapidly search optimal solutions on a low-spec computer because its source code is optimized by denoting a solution by binary code. It can solve nonlinear optimal problems as well as linear problems. DEAS is divided into two groups, local search plan and global search plan. Figure 5 represents the flow chart of the DEAS algorithm.

B. Local search strategy

DEAS is composed of Bisectional Search (BSS) and Unidirectional Search (UDS). Local search plan is optimized by using the features of a binary string. If 1 is pasted in LSB, the real number is increased, and if 0, it decreases. The BSS step is to paste 0 or 1 into the LSB of binary string and to determine the search direction. This step creates neighbor search locations at mutually opposite sides from the current search location. It can improve optimization and resolution about solution space. The USD step explores the local area until it finds the optimal value of the cost function. Figure 3 represents the flow chart of BSS. The BSS step tries to boost resolution by changing the length of the binary string and concentrates on finding the optimal solution. This step decides search locations near the current location. It initializes data except for the investigated optimal binary string in the previous section and pastes 0 and 1 at LSB of the binary string to create two neighbor search locations. The following equation (3) is applied to decode the binary string of generated neighbor search locations to be a value between 0 and 1, where *b* is the place value of the binary string and *m* is the length of the binary string. Optimal value is selected by comparing the calculated cost values. The binary string with optimal value is passed to the UDS step.

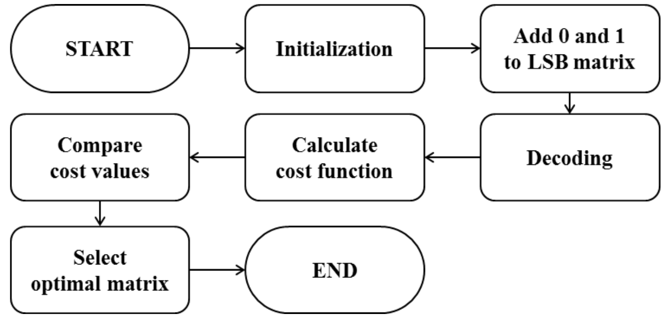


Figure 3. Bisectional search (BSS) of the DEAS

$$f_d(b_{m-1}b_{m-2}\cdots b_1 b_0) = \frac{1}{2^m - 1}\sum_{i=0}^{m-1} b_i 2^i \quad (3)$$

A cost value of a selected neighbor location is calculated by adapting the decoded value in a cost function. USD has a global search feature to search a wide range of neighbor locations, which is decided by the unmodified state (to change length of binary string from BSS).

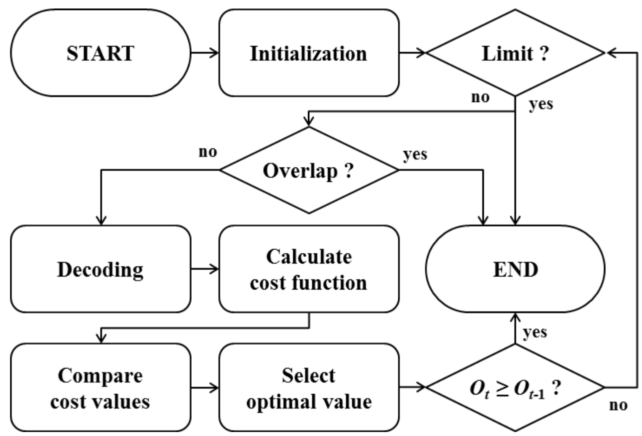


Figure 4. Unidirectional search (UDS) of the DEAS

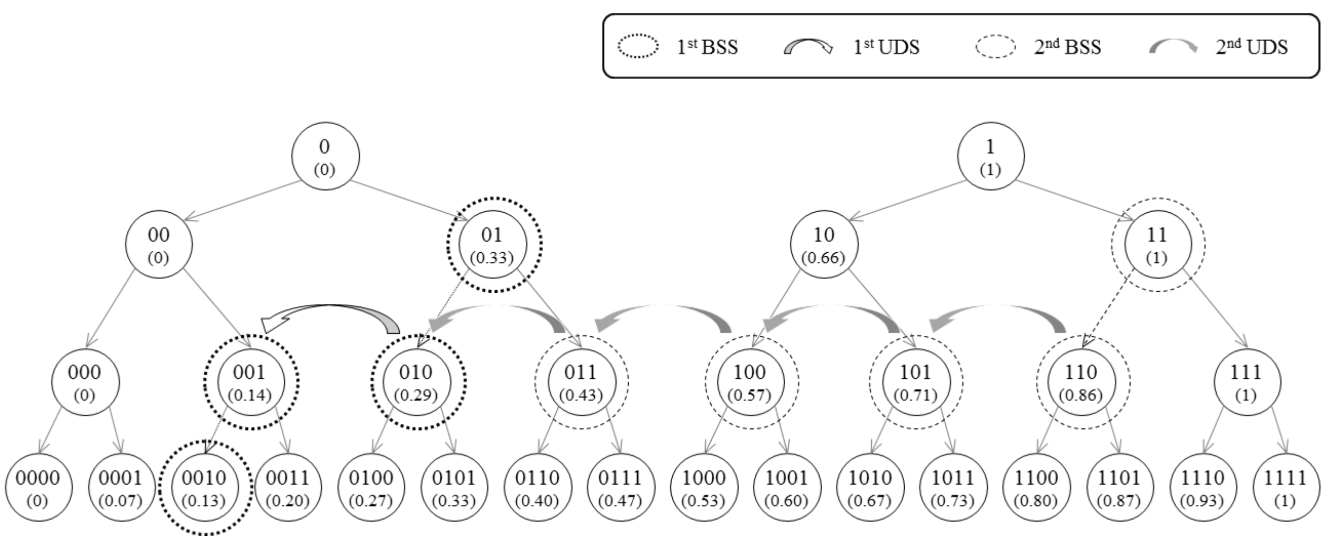


Figure 5. Flow chart of the DEAS algorithm

Also, the BSS step is just done just once but the UDS step is repetitively done until the value of the cost function is improved. Figure 4 represents the flow chart of the UDS step. In Figure 4, O_t is a currently-selected optimal solution, O_{t-1} is a previously selected optimal solution. The UDS step has two stages that are the Limit and Overlap stages. The Limit stage is a process to exclude the maximum and minimum binary strings in the UDS step. The Overlap stage is a process to exclude the previous optimal solution.

C. Global search strategy

The global search plan is performed to prevent being trapped in the local optimal solution by changing initial search locations. The global search plan can search a solution value by repetitively performing local optimal algorithms (change initial location). Figure 6 represents a flow chart of the global search plan. In Figure 6, t represents the number of performed global searches and n is the specified number of searches. In the local search plan, the searched location is stored as history to prevent searching previously searched locations.

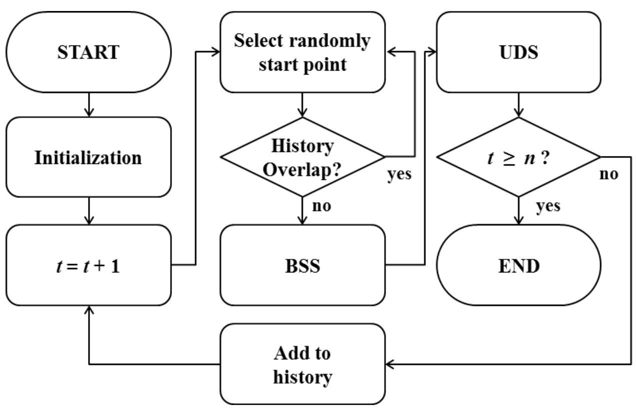


Figure 6. Global search strategy of the DEAS

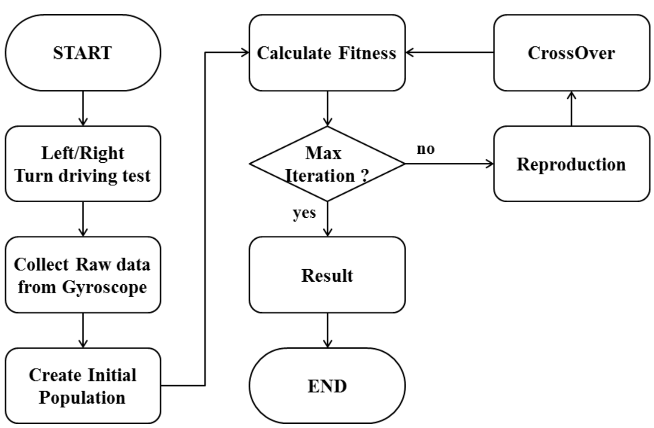


Figure 7. Sensitivitiy optimization using the GA

D. Genetic Algorithm

GA is one of the most famous algorithms in the field of optimization algorithms. Here it is used to evaluate the performance of the DEAS algorithm. GA, as stochastic search method, uses the natural phenomenon of genetic inheritance and competition for survival as its model. Figure 7 represents a flow chart of GA to optimize sensitivity of MEMS gyroscope. The repetition count number of GA is 100 in this paper. GA has crossover and reproduction operators. Two highly fitted chromosomes are selected in crossover. 50% of total chromosomes are selected as the next generation in reproduction.

E. Sensitivity value optimization of gyroscope

The proposed optimization algorithm of gyroscope's left and right sensitivities is searched through DEAS by using angle data of laser navigation after making AGV turn a full 360°. Figure 8 represents the flow chart of sensitivity optimization. *SR* is the sensitivity of right turning and *SL* sensitivity of left turning. *ESR* is a sensitivity error of right turning and *ESL* is a sensitivity of left turning. The following equation is a cost function to calculate *ESR* and *ESL*. Where O^+ is the calculated gyroscope's output through equation (4), N is the number of data.

$$f(x) = \sum_{i=0}^{N} \left(\frac{360 - x \times O^+}{N}\right) \quad (4)$$

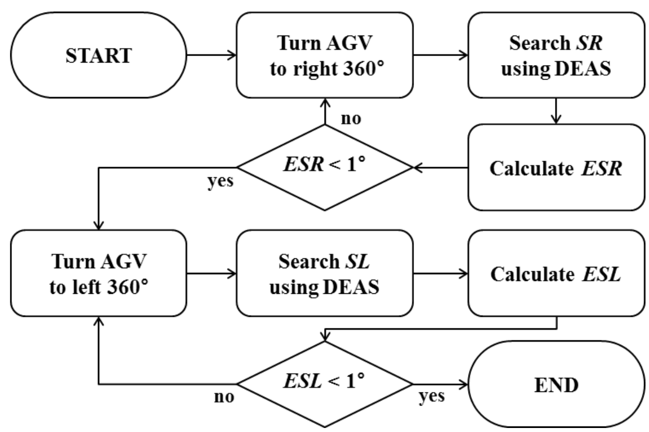


Figure 8. Sensitivity optimization using the DEAS

IV. EXPERIMENTS AND RESULTS

A. Experiment Environment

Pillar type reflectors are attached at the wall to measure the localization of laser navigation in an 840 x 2,010 *cm* experimental space. The laser navigation used in this paper is SICK's NAV200. The position can be measured with an accuracy of ±4 ~ 20 mm error. The experimental environment is Figure 9. The AGV is rotated 360° after the steer angle is fixed using laser navigation, and the raw data of the MEMS gyroscope is saved. The AGV is driven to turn left and right, each 2 times. The saved raw data of gyroscope is optimized using DEAS and GA in each turn direction. To evaluate the performance of the two algorithms, we compare the calculat-

 ISBN: 978-1-61208-352-0

ed error using the optimized sensitivity through each algorithm using the sensitivity of specification during the 360° turn.

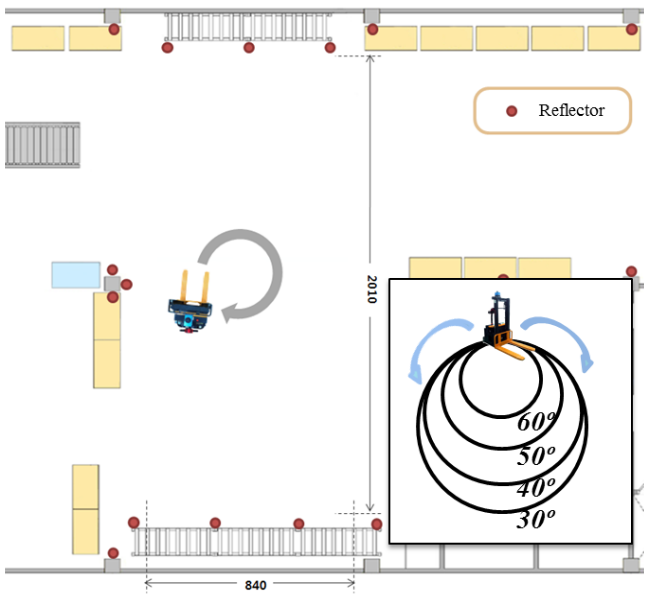


Figure 9. Experiment environment

TABLE II. RESULTS OF RIGHT TURN DRIVING TEST

Steering angle	Driving speed	Error (unit: degree)		
		DEAS 0.0250065	GA 0.0250100	Specification 0.0312000
30°	23cm/s	0.9130	1.1756	44.7037
	28cm/s	0.1366	0.3375	43.7647
	33cm/s	2.0005	1.7400	41.4367
	38cm/s	1.9141	1.6536	41.5336
	43cm/s	1.6253	1.8885	45.5025
	Average	1.3179	1.35904	43.38824
40°	23cm/s	6.3567	6.0994	36.5519
	28cm/s	7.6233	7.3669	35.1317
	33cm/s	1.2125	1.3007	44.7940
	38cm/s	1.2157	1.4785	45.0432
	43cm/s	2.3198	2.5834	46.2812
	Average	3.7456	3.7658	41.5604
50°	23cm/s	1.1268	1.0404	43.2863
	28cm/s	2.2025	1.9421	41.2102
	33cm/s	2.067	1.8609	41.3617
	38cm/s	2.6230	2.5375	42.2637
	43cm/s	0.7349	0.7930	44.1374
	Average	1.75084	1.6348	42.4518
60°	23cm/s	10.3118	10.5813	55.2430
	28cm/s	9.5866	9.8555	54.4297
	33cm/s	8.6028	8.8710	53.3266
	38cm/s	3.8603	4.1250	48.0086
	43cm/s	7.0361	7.3032	51.5698
	Average	7.8795	8.1472	52.5155
Total average		3.6734	3.7267	44.9790

B. 360° right turn driving test

Table 2 shows the angle error of the 360° rotation. The unit of sensitivity is V/°/25ms. The result of Table 2 is the average of 5 times.

The optimized sensitivity using DEAS on the right rotation was 0.0250065 V/°/25ms, and sensitivity using GA was 0.0250100 V/°/25ms. The results of error on 360° rotation are 3.4855°, 3.5347°, and 45.0002° in DEAS, GA, respectively.

TABLE III. RESULTS OF LEFT TURN DRIVING TEST

Steering angle	Driving speed	Error (unit: degree)		
		DEAS 0.0250021	GA 0.0251250	Specification 0.0312000
30°	23cm/s	1.3038	0.5964	44.3587
	28cm/s	1.6253	3.0241	45.5025
	33cm/s	1.9141	3.0223	41.5336
	38cm/s	2.0005	3.9514	41.4367
	43cm/s	0.1366	0.9397	43.7647
	Average	1.3960	2.3068	43.3192
40°	23cm/s	1.7027	2.3719	43.5772
	28cm/s	2.3198	3.6942	46.2812
	33cm/s	1.2157	0.5316	45.0432
	38cm/s	1.2125	0.4940	44.7940
	43cm/s	7.6233	1.4934	35.1317
	Average	2.8148	1.7170	42.9654
50°	23cm/s	1.1268	0.8255	43.2863
	28cm/s	2.2025	0.6864	41.2102
	33cm/s	2.0670	1.8685	41.3617
	38cm/s	2.6230	3.2094	42.2637
	43cm/s	0.7349	1.4426	44.1374
	Average	1.7508	1.6065	42.4519
60°	23cm/s	10.3118	17.8615	55.2430
	28cm/s	9.5866	12.9053	54.4297
	33cm/s	8.6028	11.9679	53.3266
	38cm/s	3.8603	10.8698	48.0086
	43cm/s	7.0361	9.3469	51.5698
	Average	7.8795	12.5903	52.5155
Total average		3.4603	4.5551	45.3130

C. 360° left turn driving test

The result of the right turn experiment in the same condition with left turn is as in the following table. As Table 3 shows, the sensitivity on the left turn using DEAS is 0.0250021V/°/25ms and the sensitivity using GA is 0.0251250V/°/25ms. The average errors on the right 360° turn driving are 3.4855°, 4.1265° on each algorithm. The average error using specification is 45.0002°. For left turn driving, less error occurred using sensitivity calculations than in using specification. This is because the sensitivity of specification does not consider the environment of the gyroscope such as tilt and electric noise, among others.

Figure 10 shows error using optimized sensitivity through each algorithm. As verified in Figure 10, the left part in the boxplot is the experiment result (angle error) of the

 ISBN: 978-1-61208-352-0

right turn driving test and the right part is the experiment result of the left turn driving. In the result, errors using DEAS and GA are lower than errors of specification values from the left/right turn-driving experiment. However, GA is not proper in MCU because it requires high computation to show good performance. DEAS is proper in MCU, which finds the optimal solution of cost function by only using well-planned operations.

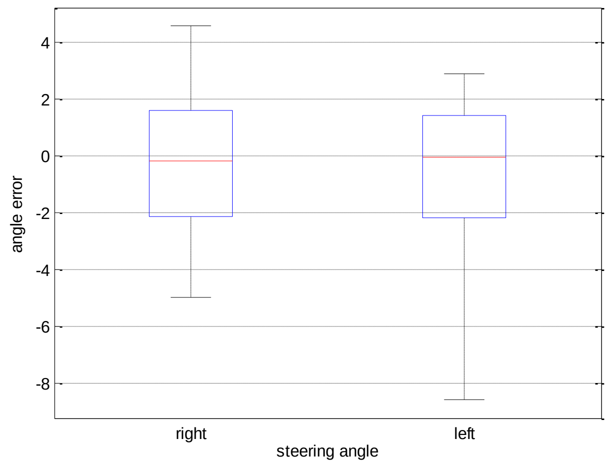


Figure 10. Boxplot of errors on right/left turn driving tests

V. CONCLUSION

In this paper, we implemented and compared performances using DEAS and GA, which are well known optimization methods. Sensitivity optimization of MEMS gyroscope is very important due to low angular velocity accuracy and a low straightness performance. Therefore, a sensitivity value of a gyroscope should be optimized from MCU to measure the data of a gyroscope. Because existing optimization methods have high computation and complex algorithms, it is difficult to apply in MCU. However, a well-planned DEAS algorithm is simple and has a lower computation time than the existing algorithm. Therefore, this paper suggests sensitivity optimization of a gyroscope through DEAS. To verify the performance of the proposed method, we compared the result of DEAS with the result of GA. As for experimental results, the average error using sensitivity of specification (in gyroscope datasheet) was 45.0002°, 45.0002° on turns right and left, respectively. The average error of sensitivity value through DEAS and GA from right turn driving respectively are 3.4855°, 3.5347°, and the result of left turn driving are 3.4855°, 4.1265° respectively. As calculated error using optimized sensitivity through DEAS and GA, average errors on right turn driving are 0.0250065°, 0.0250100° and on left turn driving is 0.0250021°, 0.0251250°. As the results show, the proposed method had a better performance than the result of specification. As part of future research, we plan to study what reduces drift and changes sensitivity in real-time with tile of AGV.

ACKNOWLEDGMENT

This research was supported by the MOTIE, KOREA under the Human Resources Development Program for SPENALO Research Center support program supervised by the NIPA (H1502-13-1001) and supported by a 2-Year Research Grant of Pusan National University.

REFERENCES

[1] M. P. Groover, “Automation, Production Systems and Computer Integrated Manufacturing”, sigma press, August, 2007.

[2] J. Ota, A. Shinozaki and H. Hashimoto, “Hybrid Design Methodology and Cost-Effectiveness Evaluation of AGV Transportation System,” Automation Science and Engineering, IEEE Transaction on, vol. 4, no. 3, July, 2007, pp. 360-372.

[3] J. Villagra and D. Herrero-Perez, “A Comparison of Control Techniques for Robust Docking Maneuvers of an AGV,” Control Systems Technology, IEEE Trans. on, vol. 20, no. 4, July, 2012, pp. 1116-1123.

[4] C. Y. Chan, “A System Review of Magnetic Sensing System for Ground Vehicle Control and Guidance,” California PATH Research Report, UCB-ITS-PRR-2002-20, May, 2002.

[5] Y. Pang, “Bipolar magnetic positioning system for automated guided vehicles,” Intelligent Vehicles Symposium 2008 IEEE, June, 2008, pp. 883-888.

[6] J. W. Kim and S. W. Kim, “Parameter identification of induction motors using dynamic encoding algorithm for searches,” IEEE Transaction on Energy Conversion, vol. 20, no. 1, March, 2005, pp. 16-24.

[7] J. W. Kim and S. W. Kim, “Numerical method for global optimization: dynamic encoding algorithm for searches,” IEE Proceeding Control Theory and Applications, vol. 151, no. 5, September, 2004z, pp. 661-668.

The Concept of Attack Surface Reasoning

Michael Atighetchi, Nathaniel Soule, Ron Watro, Joseph Loyall
Raytheon BBN Technologies
Cambridge, MA
{matighet, nsoule, rwatro, jloyall}@bbn.com

***Abstract*—Today's cyber defenses and cyber defenders face determined and diverse adversaries, who can study all aspects of deployed systems including networks, hosts, and the applications running on them, in order to find exploitable vulnerabilities and to devise attack vectors that exploit the detected vulnerabilities. The conflict between cyber attackers and cyber defenders is stacked against the defender. The defender must protect against all the ways that an adversary can cause potential loss of security, collectively called the *attack surface*, while the attacker needs to find only a single vulnerability and attack vector to be successful. This work-in-progress paper describes an AI-inspired approach for modeling and analyzing the attack surface of a distributed system. Once modeled, an attack surface can be quantified in terms of size and level of dynamism through four types of algorithms: path analysis, metric computation, path comparison, and path enumeration. Our approach supports relative comparison across multiple attack models for each combination of a system and a set of defenses, in order to select an appropriate set of defenses given a certain cost/benefit tradeoff.**

Keywords: security, semantic web, analytical models

I. Introduction

Today's cyber defenders face determined, diverse, and well-resourced adversaries who have a significant advantage over the defense. The adversaries need find only a single vulnerability and attack vector to be successful, while the defender must protect all vulnerabilities and defend against all attack vectors. Among the various choices of available defenses, a powerful class is the set of Moving Target Defenses (MTDs) that attempt to even the playing field for defenders by shifting the attack surface, i.e., the set of potential attack vectors an adversary can use to compromise security of a target system. MTDs attempt to change access paths before they can be exploited, thereby (1) making them more difficult to detect, (2) rendering attacks that are based on stale information ineffective, and (3) increasing detection by monitoring for the use of stale information.

However, as the number and complexity of defenses (including MTDs), system configurations, and potential attacks continually increase, cyber defenders face the problem of manually selecting and configuring defenses for a distributed mission-critical system without a clear understanding of the seams/integration points, residual risks, and costs (in terms of impact on performance and functionality). Integration of defenses performed in a non-structured way bears the risks of adding defenses with no value, inadvertently increasing the attack surface, or overly impacting critical functionality.

This paper describes work in progress for constructing a model-based environment for *Attack Surface Reasoning (ASR)*, i.e., integrating and evaluating systems and defenses and analyzing compositions of systems and dynamic defenses.

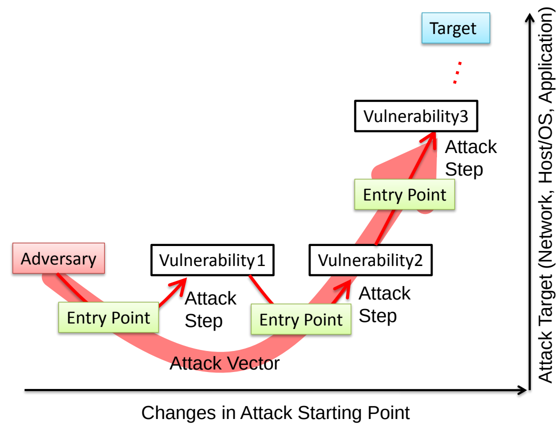


Fig. 1. Attack surface models include the starting points and the targets of attacks, which can exist at multiple horizontal and vertical layers, respectively.

The models, metrics, and algorithms used in measuring attack surfaces need to support the following key requirements:

The attack surface model needs to represent concepts and defenses situated at multiple layers. Attacks may target resources at the network layer (both network traffic observed on intermediary network components between the client and the server as well as the Transmission Control Protocol (TCP)/Internet Protocol (IP) stacks of servers and clients), the operating system and host layers (e.g., by running attacks against a Java Virtual Machine (JVM) compute platform), and the application layer (e.g., by corrupting Structured Query Language (SQL) tables). We refer to this requirement as *vertical layering*, as visualized by the attack vector in Fig. 1 as it moves from Vulnerability 2 to Vulnerability 3.

The attack surface model needs to capture ordering dependencies between control and data flows and defenses to be employed against attack vectors. Attacks consist of an orchestrated execution of individual attack steps, where the effectiveness of each attack step is contingent on the starting point (and level of privilege) in relation to the specific target. For instance, network attacks launched from a legitimate client have a significantly different starting point compared to the same attacks launched from an adversary-controlled device attached to the network. This knowledge should be part of the attack surface model. An example of ordering at a given vertical layer includes a firewall placed in front of a protected host, requiring all traffic to pass through the firewall first before reaching the host's TCP/IP stack. We refer to this requirement as horizontal layering, as visualized by moving from Vulnerability 1 to Vulnerability 2 in Fig. 1.

 ISBN: 978-1-61208-352-0

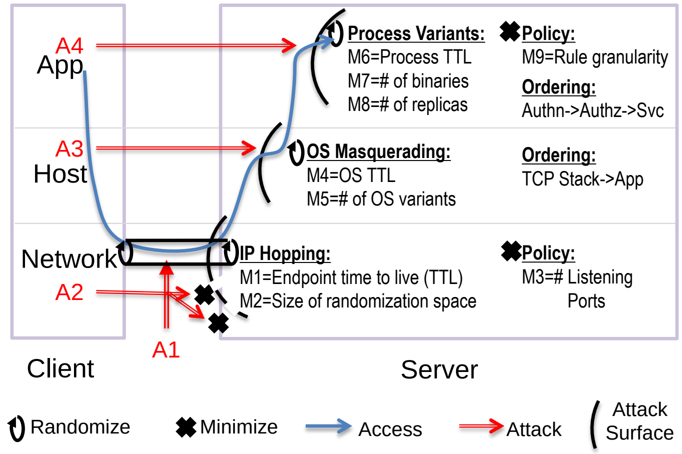


Fig. 2. Multi-layered attack surface randomization and minimization

The attack surface models need to capture complexities associated with dynamic defenses. Many of the attack effects described so far are difficult to detect automatically and cause observable events only when critical mission functionality starts failing, leaving little time for recovery and increasing the impact of the attack. While proactive shaping strategies, such as IP Hopping, Operating System (OS) masquerading, unpredictable replication, and Address Space Layout Randomization (ASLR) are clearly beneficial, the existence of multiple dynamic adaptations at different system layers makes management of the overall policy and configuration sets deployed on the target devices challenging.

Fig. 2 shows an example attack surface, consisting of a single Server composed with three MTDs: (1) IP Hopping that randomizes IP addresses between the client and server at the network layer, (2) OS masquerading that changes parameters in the OS stack to make one OS look like another, and (3) Process variants that generate different functionally equivalent versions of binaries and replicas. The overall goal of ASR is to quantify the attack surface along two main attributes – minimization and randomization – with the objective of minimizing access where possible and randomizing information about remaining access where affordable.

The algorithms to compute minimization rely on the system and mission models and metrics to determine how much of the attack surface really needs to be exposed to effectively function or, conversely, the parts of the attack surface that can be reduced or removed without adversely affecting critical functionality. For example, some applications with external facing interfaces (thereby providing entry into the system if penetrated) might not be needed during some phases of operation – or might not be as critical – and can be shut down, removing those entry points into the system and their associated vulnerabilities and attack vectors.

The algorithms to compute randomization execute over the models and metrics for a system deployment and a set of defenses, including MTDs, and compute how the defenses or combination of defenses change the attack surface.

The paper is structured as follows. Section 2 describes related work. Section 3 describes a high-level view of attack surface models. Section 4 describes algorithms for quantifying attack surface models. Section 5 concludes the paper.

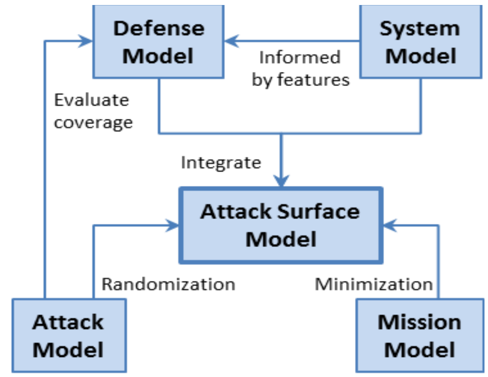


Fig. 3. Construction of the Attack Surface Model

II. RELATED WORK

Our work relates to several efforts in the areas of cyber vulnerability modeling, light-weight formal methods, and mathematical attack surface definitions.

The Mitre Common Vulnerabilities and Exposures (CVE) [1] database maintains a comprehensive list of specific vulnerabilities that can be used to establish the attack model. Our definition of an attack surface extends the mathematical models presented in [2] by including vertical and horizontal layering as a central property of an attack surface. Decision support analysis systems for cyber defense such as [3] employ probabilistic techniques by annotating attack trees with defense information. This work focuses less on determining attack success probability over a given system, instead focusing on supporting comparative and recommendation based analytics. Performing quantitative reachability analysis has been studied extensively in the academic literature, using domain-specific languages such as Alloy [4], Lobster [5], or Cross Domain Entitlement Language (CDEL) [6]. The algorithms described in this paper operate on standards-based semantic web models rather than models described in proprietary languages. Our semantic web approach, using Web Ontology Language (OWL) [7], allows disparate sources of information to be automatically integrated [8] into a graph of interconnected information that is easy to understand and extend because it is based on real world concepts.

III. ATTACK SURFACE MODEL

Modeling an attack surface involves linking several different models, each representing aspects of the defended system. An attack model, describing goals of an attack, together with starting points, can be used to evaluate the attack surface for randomization attributes. The mission model, describing the set of required interactions in support of mission critical functionality, can be used to minimize access by pruning unnecessary access paths and highlighting those elements most important to mission success. Fig. 3 shows how a defense model and system model (shown in the top) together build the main ingredients of an attack surface model.

Fig. 4 displays a high-level ontological view of the defense model. It shows how a defense is described by a *setup* over a set of resources. For IP hopping, the setup contains the set of machines that have access to the hopping scheme's secret key and can therefore compute what the next IP address is going to

 ISBN: 978-1-61208-352-0

be. Attacks that start from the hosts that are part of the setup are not impeded in any way by the defense, since they all have access to the secret key of the hopping scheme and therefore perceive no randomization. Fig. 4 also shows that a defense provides protection for a set of resources, as expressed by the *Defense → provides → Protection* link. We model the type of protection provided in terms of overall security benefit, including confidentiality, integrity, availability, and discoverability. Furthermore, we express the mechanism of protection that the defense imposes. The *Protection → through → Filtering* link captures any security check that explicitly drops data, e.g., as it is being flagged by anomaly detection at various layers. The *Protection → through → Randomization* link captures aspects of dynamism associated with proactive shaping strategies. As shown in Fig. 4, the protections directly link to the scope of the defense, in terms of entry points, exit points, and data, which are key aspects of our definition of the attack surface. Finally, each defense has a cost associated with it (as shown to the right of Fig. 4). We include the fact that defenses can increase the attack surface via *Defense → adds → Data* and *Defense → adds → Point* links.

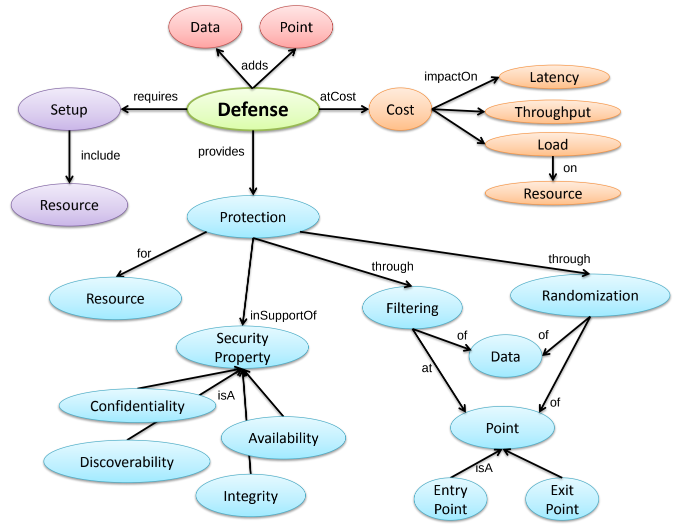


Fig. 4. Ontology representation of a generic dynamic defense

IV. QUANTIFICATION OF ATTACK SURFACES

The randomization, minimization, and other characterization metrics and analyses provided by ASR share an underlying common base of feasible-path analysis, but can be classified into distinct groups based on the operations that occur post-path-determination.

Metric Computation allows for calculation of metrics describing the randomization, minimization, or other characteristics of a point, path, or system, including the metrics in Tab. 1.

Path Comparison calculations execute path differencing and comparison algorithms to determine the set of elements in a system that are not required to support a given mission, and the disabling of which will help reduce the attack surface without degrading operation.

Path Enumeration analyses are undertaken to discover points, paths, and system configurations that exhibit certain properties, such as *all paths that contain defenses with dynamic frequencies less than 5 minutes*.

Fig. 5 depicts an example system model and defense model which, when integrated, form an attack surface model, along with an accompanying simplified attack model. The system model, shown in blue in the center, describes a single host with two network interface cards (NICs), through which a single service may be accessed. The service is also exposed internally on the host via an Application Programming Interface (API). A single defense has been modeled in the purple nodes at the top, in this case an IP hopping MTD named DYNAT [9]. The MTD has been configured to protect the IP address of *NIC 1* on *Host 1*. An attack class, shown in red at the bottom, has been modeled (in an abbreviated form) describing an attack category that is relevant to network endpoints. Given a model such as this, ASR's processing engine allows for the performance of many interesting analyses and metric computations. For example, in order to determine how many paths are protected by defenses that change more slowly than some known attack class's expected duration (and are thus not dynamic enough to provide robust protection), one may compute the metric *number of defenses with dynamic modulation frequencies greater than any applicable attack phase duration*. Calculation of this metric is accomplished by first performing a path analysis identifying all possible paths through the system – for this example binding the starting points to network entry points (thus ignoring insider threats) based on the given attack model domain, and binding the goal states to all "services." Given these starting points and goals, six possible attack paths exist. Fig. 5 shows three of these paths (paths 1, 2, 3), with the other three being symmetric mirrors, starting from *Network EP* 2. The ASR path analysis engine will identify the paths starting from the two network entry points (*Network EP 1*, and *Network EP 2*) and branching from there to go directly to Service 1 (via path 1 and its symmetric equivalent), indirectly to Service 1 through the API(via path 2 and its symmetric equivalent), or indirectly through the opposing NIC (via path 3 and its symmetric equivalent). At this point the identified paths are stored such that they may be reused for multiple queries in the second phase of analysis.

The second and third phases for the metric in question in-

TABLE I. HIGH-LEVEL METRICS

Randomization
Number of defenses with dynamic modulation frequencies greater than any applicable attack phase duration
Defense modulation frequency
Number of dynamic surfaces
Number of dynamic surfaces per protection type (confidentiality, integrity, availability, discoverability)
Minimization
Number of deployed defenses
Attack surface area change due to defenses
Number of defense boundaries crossed per path
Number of paths with less than N defenses
Rule granularity of defenses
Number of processes
Number of open ports
Number of users
Other Characterizations
Sum of path lengths from entry point to each defense
Is there at least 1 defense per known attack class
Number of paths with conflicting defense types
Number of entry points without a defense within 1 hop

volves executing SPARQL Protocol and RDF Query Language (SPARQL) queries (to identify all defenses meeting the prescribed criteria) and aggregation operations (here, a simple sum) over the set of paths. Three of the feasible paths (paths 1, 2, and 3) encounter an MTD whose randomization frequency is on the order of seconds. Since the attack in the simplified attack class model has an expected duration on the order of minutes, the MTD is determined to provide adequate protection (for the paths it covers) and thus this metric will have a value of 0, i.e., there are no defenses with dynamic modulation frequencies greater than the attack phase duration. However, determining the overall protection of this system should evaluate this metric in the context of other important metrics. For example, calculating the metric *number of paths with fewer than N defenses* would highlight that entry through *NIC 2* is unprotected. Further, had the metric been defined to include a larger set of starting points the resulting value would have been even larger, as insider attacks that start from a privileged base on *Host 1* will not pass through any defense.

In addition to the base path exploration, many other analyses may be performed. For example, the user may wish to elaborate on the numeric value calculated as part of the *number of paths with fewer than N defenses* metric by drawing from the enumeration category of analyses to ask ASR to ***identify*** *all paths that include less than one defense*. Again, a SPARQL query is defined to operate over the initial set of feasible paths, and select only those that include no defenses.

V. CONCLUSION AND NEXT STEPS

This paper describes work we are conducting on modeling and reasoning about attack surfaces of distributed systems, with the goal of selecting a properly configured set of defenses that together minimize access where possible and randomize observables where feasible. Our approach enables quantification of the attack surfaces for the purpose of performing relative comparisons between multiple surfaces, each one representing a set of defenses, system components, and attack classes.

In future work, we plan to develop and evaluate example scenarios, involving tradeoffs between multiple defenses at different layers, some providing great value in minimizing and randomizing the attack surface while others actually increase the attack surface or negatively impact availability by introducing excessive dynamism or cost.

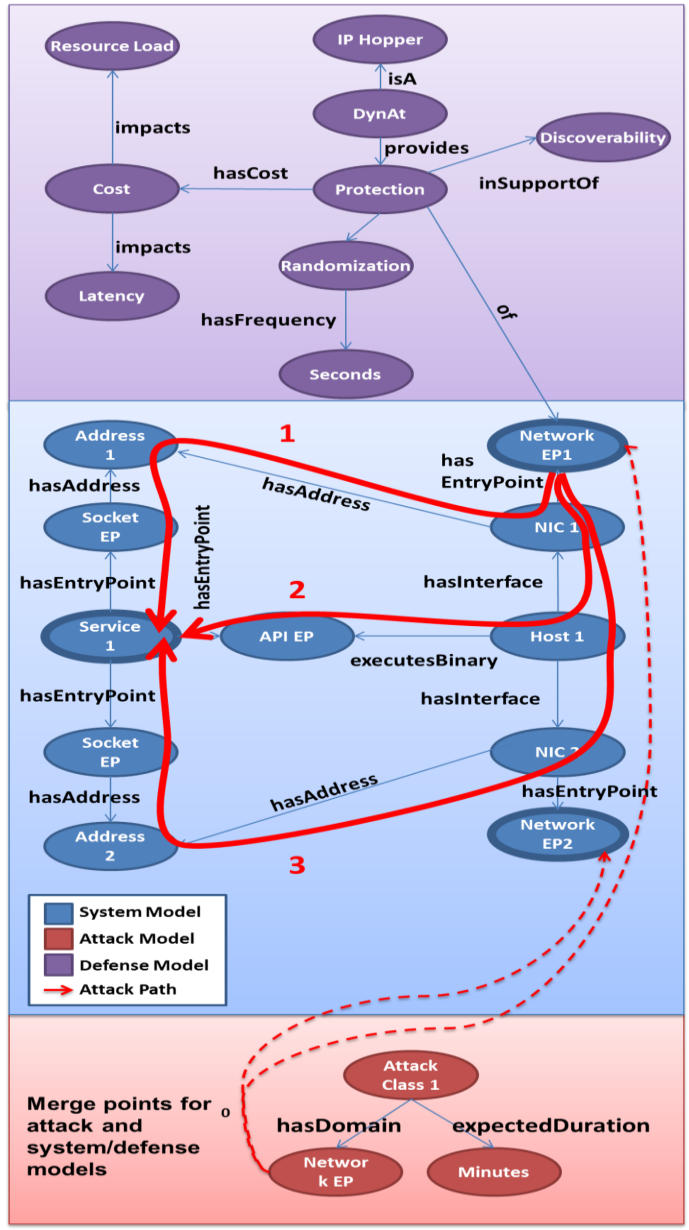


Fig. 5. Example attack surface model used for analysis

REFERENCES

[1] Mitre, "Common Vulnerabilities and Exposures Home Page," 2014. [Online]. Available: http://cve.mitre.org/. [retrieved: 2014.03.11]

[2] P. K. Manadhata and J. M. Wing, "A Formal Model for a System's Attack Surface," in Moving Target Defense, Springer, 2011, pp. 1–28.

[3] T. Sommestad, M. Ekstedt, and P. Johnson, "Cyber Security Risks Assessment with Bayesian Defense Graphs and Architectural Models," presented at the Hawaii International Conference on System Sciences (HICSS '09), Hawaii, 2009, pp. 1–10.

[4] D. Akhawe, A. Barth, P. E. Lam, J. Mitchell, and D. Song, "Towards a formal foundation of web security," in 23rd IEEE Computer Security Foundations Symposium (CSF), 2010, pp. 290–304.

[5] J. Hurd, M. Carlsson, B. Letner, and P. White, "Lobster: A domain specific language for SELinux policies," Galois Inc. internal report, 2008.

[6] J. Beal, J. Webb, and M. Atighetchi, "Adjustable autonomy for cross-domain entitlement decisions," in Proceedings of the 3rd ACM workshop on Artificial Intelligence and Security, 2010, pp. 65–71.

[7] D. L. McGuinness, et al., "OWL web ontology language overview," W3C Recomm., vol. 10, no. 2004–03, p. 10, 2004.

[8] M. Fisher and M. Dean, "Semantic Query: Solving the Needs of a Net-Centric Data Sharing." Semantic Technology Conference, 23-May-2007.

[9] D. Kewley, R. Fink, J. Lowry, and M. Dean, "Dynamic approaches to thwart adversary intelligence gathering," DARPA Information Survivability Conference & Exposition (DISCEX), 2001, vol. 1, pp. 176–185.

 ISBN: 978-1-61208-352-0

A Method for Evolutionary Decision Reconciliation, and Expert Theorems

Vladislav Protasov, Zinaida Potapova, and Eugene Melnikov
Center of Computing for Physics and Technology
Moscow, Russia
protvlad@gmail.com, zinaida.potapova@gmail.com, apinae1@gmail.com

Abstract—**The author believes that the lack of a theory behind collective intelligence systems and efficient information technologies is one of the reasons for slowing down the development of e-democracy processes. Existing government system, the democracy of the voting majority, is based on ideas that are more than 200 years old and on the well-known Condorcet's jury theorem. One of this theorem's corollaries is that a decision made by the majority vote of a group is worse than a decision made by a single member of the same group if the members of this group make incorrect decisions more often than they make correct decisions. This paper provides proof of two expert theorems proposed by the author. These theorems apply to groups of experts making joint decisions, and they propose that a group of experts involved in reconciliation increase the probability of correct decision if the experts have an opportunity to select and vote on the best third-party decisions. Conditions are provided under which the probability of correct decision by the group of experts approaches 1 as the number of the experts increases. These theorems indicate the direction for development of network programs helping groups of low-competence experts to overcome the Condorcet's border.**

Keywords- collective intelligence systems; Condorcet's jury theorem; e-democracy; crowdsourcing; evolutionary decision reconciliation.

I. INTRODUCTION

The development and use of collective intelligence systems is a popular trend, with individual intelligence no longer being able to keep the pace with the development of our civilization. More than 700 crowdsourcing platforms have been developed and are in use today [1]. Some of them are used directly in e-democracy initiatives [2]. There is an opinion that development of network information technologies will inevitably lead to the emergence of planetary intelligence and the new type of public government—direct e-democracy [3]. The author believes that the lack of a theory behind collective intelligence systems and efficient information technologies is one of the reasons for slowing down this process. Existing government system—the democracy of the voting majority—is based on more than 200-year-old ideas and works by Marquis de Condorcet [16], and no longer fits the spirit of the times. Modern times require a more progressive democracy system relying on the latest network technologies and, most importantly, the new organizational principles for collective intelligence systems.

Collective intelligence is a term first used in mid-1980s in sociology in research of collective decision-making process. NJIT (New Jersey Institute of Technology) researchers defined collective intelligence as the ability of a group to find solutions to a problem that are more efficient than any of the solutions found by individual members of the same group[4]. This concept is used in sociobiology, political science, group reviewing and crowdsourcing applications[5]. It can also be defined as the product of collaboration between the people and data processing methods. In this definition, collective intelligence is referred to as "symbiotic intelligence" and is described by Norman Lee Johnson[6]. According to Lévy, this phenomenon is related to the ability of network information and communication technologies to expand the common body of social knowledge and the range of possible collaboration among people [7].

As described in [8], certain factors related to challenges in human interactions often make a consolidated group decision unreachable, and these environments require the use of efficient coordination methods for group collaboration and the involvement of facilitators in charge of such coordination.

A method for evolutionary decision reconciliation currently researched in Russia is, to a large degree, free of these limitations [9][10]. Specially designed rules for interactions between group members based on genetic algorithms are used as facilitators.

This method can be briefly described as follows. A group of experts receives a problem with a clear goal and clear requirements for solution, which should be presented in text form. Experts work anonymously and interact by exchanging partial solutions with one another via a computer network.

The first stage is solution generation stage when experts create variants of partial solution to the problem based on the project goal. The second and further stages are iterative reconciliation stages when experts evaluate others' solutions and select what they believe to be the best parts of these solutions. Number of variants to be evaluated depends on the interaction rules. Others' solutions for reconciliation are chosen randomly, just like in genetic algorithms. Iterations continue until the allocated time runs out or more than half experts end up with identical solutions. In the first case, group solution is created as a combination of parts of individual solutions with the largest number of matches. In the second case, group solution is the solution chosen by the majority of experts.

The paper is structured as follows. The next section after introduction describes the new technology developed by the authors that is based on the Evolutionary Decision Reconcillation method. In the third section the Condorcet's theorem is described with discussing the conditions of obtaining the solution with probability, close to one, for crowdsourcing systems.The fourth section presents and proves basic theorems of the new technology that allows experts to overcome the "Condorcet's border". In the fifth section possible areas of the technology application are given. The sixth section concludes the paper.

II. METHOD FOR EVOLUTIONARY DECISION RECONCILIATION

This paper discusses several recent results that, in the author's opinion, may support the principles of metasystem transitions offered by Turchin [11]. Groups of individuals and basic computers were used as components of a system representing the next-generation intelligence. This system demonstrated considerably higher "intelligence" than the combined intelligence of its isolated constituents.

Proposed approach can prove useful in the development of artificial intelligence. This method could help to design symbiotic architectures from neuronets and neurocomputers, computers and their networks, groups of people and genetic rules working as a single unit. Protasov successfully applied these rules to improve human intelligence [12] and the "group intelligence" of robot groups [13]. A similar approach could perhaps be used to construct hierarchical networks with cascaded "intelligence" gains at each level.

For example, a popular target distribution problem can be formulated as follows: there are m robots with calculators and n targets. Each calculator has coordinates of all robots and all targets. The goal is to split robots and targets in pairs so that if $m < n$ or $m = n$, then there is at least one robot for every target, and if $m > n$, then there is at least one target for every robot, and the sum of distances S between robots and their paired targets is minimal.

Experience shows that the generic method is sufficiently effective in solving this type of problems. The most trivial solution would be as follows: since the every calculator of every robot i has location data for all robots and targets, it can build a set of possible target distributions using standard crossover, mutation, estimation and selection operations, through some iterations will result in less than optimal solutions. Since all calculators use the same algorithm and have identical data sets, they generate identical solutions when they operate independently, and each robot will choose its target. The advantage of the proposed method is that it can be used to speed up the calculations by a factor of m by making a more efficient use of resources.

In [13], a method for distributed calculations was proposed whereby one super-calculator coordinates the work of other calculators to reduce the overall time required to make a decision.

For the target distribution problem, the following algorithm was proposed: each calculator will use its own random number generator to create one possible solution, which will likely be far from optimal. All proposed solutions will be different. Then the calculators will communicate and exchange proposed solutions. After that, the calculators reject the worst half of the overall number of solutions, certain mutations in proposed solutions, and exchange solutions again. This process will continue until only one solution remains. Then the calculations will stop, and each robot will have the best solution in its possession.

A demo program called COLLINTROB was developed and tested on Delphi platform. It demonstrated that the time to reach a common decision was inversely proportional to the number of calculators in system. Some experiments simulated failures of a certain number of calculators and demonstrated that calculator failures did not degrade the quality of solution (the result was within 5% from the ideal solution), but the time required to reach the decision increased proportionally $M / (m-k)$, where k is the number of failed calculators.

The analysis of these experiments suggests that the "collective intelligence" of calculators incorporated in the super-calculator increases and it becomes greater than the intelligence of individual calculators on a certain class of problems. In other words, in this case we see the metasystem transition resulting in the occurrence of "greater mind" consisting from artificial components with a lower level of "intelligence".

The same rules of interaction between the individual components of such a "collective mind" were used in tests using human subjects and were applied to create the new kind of collective intelligence in the so-called Method for Evolutionary Decision Reconciliation (MER) [9].

The variables, such as the number of participants or the number of proposals discarded at each step, can vary depending on the type of problem and can be selected experimentally.

The Method for Evolutionary Decision Reconciliation was tested in groups of 4 to 20 male students in different areas including collective poetry, music composition, creation of psychological portraits, development of simple computer program, selection of the best move in a chess game, direct sales, portrait painting, and creation of abstract diagrams.

These experiments confirmed that collective intelligence exceeds the combined intelligence of individual contributors and shows the phenomenon of knowledge transition from the strongest contributors to the weakest. This method also allows ranking the contribution of individual participants to the final product. It usually did not take long to solve simple tasks, and as a rule, the duration of experiments did not exceed two hours, with Eysenck tests taking approximately 30 minutes). Students were quite enthusiastic about these experiments, and excited about being a part of collective mind exceeding their individual capabilities.

III. CONDORCET'S JURY THEOREM

Protasov et al. [14] provide the results of computer modeling for a collective decision-making process using the new technology. Certain specific competence levels of experts, who generate ideas and evaluate others' solutions, were shown to amplify the intelligence when the experts were working with the group. This paper contains the proofs of theorems confirming this effect.

Let us start with discussing the limitations of collective intelligence systems following from Condorcet's jury theorem. One of the definitions of Condorcet's jury theorem [15] is as follows:

Let us assume that one of the two decisions proposed to the jury is correct, and each jury member, on average, makes correct decisions more often than not. The theorem claims that as the number of participants increases, the probability that the correct decision is made tends to one.

The probability that the group of *M* experts makes correct decision, assuming that each of these experts makes correct decision with the probability of *G*, can be calculated as follows:

$$K_0 = \sum_{i=0}^{\frac{M-1}{2}} \binom{i}{M} G^{M-i}(1-G)^i \qquad (1)$$

With $G>0.5$ and $M \to \infty$, we have the Condorcet effect, and the probability of correct decision by the group of experts $K_0 \to 1$, but with $G<0.5$, the probability $K_0 \to 0$. In other words, $G=0.5$ is the border value that weak experts are unable to overcome.

The Condorcet's effect is used in modern crowdsourcing systems where tens and hundreds of thousands of users find the best decisions. Unfortunately, G is not guaranteed to be always greater than 0.5. We will show that if we give experts with 0<G<0.5, who did not make their decisions for any reason, an opportunity to view decisions of other people and select the best of them, then under certain conditions the probability that a group of such experts makes a correct decision by majority vote tends to 1 with the increase of the number of experts in the group, or in other words, such a group is able to overcome the Condorcet's border.

The purpose of this research is to evaluate the competence of experts that guarantees that the probability that the group of experts makes the correct decision approaches 1 as the number of experts increases.

Experts are supposed to use evolutionary decision reconciliation in their collective work.

This work presents proofs for two theorems that help to forecast the results of group effort based on expert competence levels, offers several corollaries and discusses benefits and use cases of the new technology.

IV. EXPERT THEOREMS

A. Theorem 1

Let us assume that at the individual decisions stage, each expert makes correct decision with the probability $0 < G_P < 0.5$, incorrect decision with the probability $0 < G_N < 0.5$, or no decision with the probability $G_V = 1-(G_P + G_N)$, and at the group decisions stage, an expert that did not make any decision selects correct decision out of several third-party decisions with the probability E_P (assuming it is available), incorrect decision with the probability E_N, or no decision. The theorem proposes that when $G_P + E_P \frac{1-(G_P+G_N)}{E_P+E_N} > 0.5$ condition is met, the probability that correct decision is selected by majority vote at reconciliation stage increases and tends to one as the number of experts increases.

Proof. At the individual decisions stage, the expected value of the number of experts P_0 who make the right decision (subgroup *P*) is $G_P M$ (where M is the total number of experts); the expected value of the number of experts N_0 who make the wrong decision (subgroup *N*) is $G_N M$; and the expected value of the number of experts V_0 that make no decision (subgroup *V*) is $G_V M = (1-G_P-G_N)M$.

At reconciliation stage, experts from subgroup *V* will join experts in subgroups *P* and *N* in proportion to E_P and E_N, while subgroup V will reduce in proportion to $E_V = 1-E_P-E_N$. Let us designate P_i, N_i and V_i as the expected values of the number of experts in respective subgroups at the i-th iteration of the reconciliation stage.

Iteration 1: $P_1 = (G_P + E_P G_V)M$, $N_1 = (G_N + E_N G_V)M$, $V_1 = E_V G_V M$.

Iteration 2: $P_2 = (G_P + E_P G_V + E_P E_V G_V)M$, $N_2 = (G_N + E_N G_V + E_N E_V G_V)M$, $V_2 = E_V^2 G_V M$.

…

Iteration i: $P_i = G_P M + E_P G_V M \sum_{k=0}^{i-1} E_V^K$,

$N_i = G_N M + E_N G_V M \sum_{k=0}^{i-1} E_V^K$, $\qquad V_i = E_V^i G_V M$.

With $i \to \infty$, if we replace $\sum_{k=0}^{i-1} E_V^K$ with its limiting value $\frac{1}{1-E_V}$ and the last term of geometric progression V_i with zero, we will get the expected values P_B, N_B and V_B for the numbers of experts in groups *P*, *N* and *V*:

$$P_B = (G_P + \frac{E_P G_V}{1-E_V})M \ , \ N_B = (G_N + \frac{E_N G_V}{1-E_V})M \ , \ V_B = 0 . \qquad (2)$$

With $G_P + E_P \frac{1-(G_P+G_N)}{E_P+E_N} > 0.5$, the majority of the group will make the right decision. Since with $M \to \infty$, the P_B / M ratio is the probability that the expert makes correct decision at the end of reconciliation stage, the condition of Condorcet's theory is met, and our theorem is therefore proven.

For practical use and theoretical research, the condition of this theorem can be expressed as

$$\frac{E_P}{E_N} > \frac{1-2G_P}{1-2G_N}. \quad (3)$$

Let us review several corollaries of (3):

If the experts in the group have weak decision-making capability ($G_P < G_N$), then for the probability that they make correct decision to be more than *0.5*, they need strong evaluation capability ($E_P > E_N$).

If the experts in the group have strong decision-making capability ($G_P > G_N$), then for the probability that they make correct decision to be higher than *0.5*, even weak evaluation capability is sufficient ($E_P < E_N$).

If both capabilities are weak in the group, experts will not be able to make correct decision with probability higher than *0.5*. Moreover, the probability that the vote results in correct decision decreases and tends to zero as the number of such experts increases.

B. Theorem 2

Suppose that at the individual decision stage, every expert makes correct decision with the probability of $0 < G_P < 0.5$, incorrect decision with the probability of $0 < G_N < 0.5$, or no decision with the probability of $G_V = 1-(G_P+G_N)$; and at the first iteration of reconciliation stage, each expert who did not make a decision receives a randomly selected third-party decision, and then the expert correctly evaluates the correctness of this decision with the probability of E_R, and as a consequence, either submits or does not submit this decision for a vote.

The theorem proposes that when $E_R \frac{1-(G_P+G_N)}{G_P+G_N} > \frac{1-2G_P}{2G_P}$ condition is met, the probability that correct decision is selected by majority vote at the first iteration of reconciliation stage increases and tends to one as the number of experts increases.

Proof. At the individual decisions stage, the expected value of the number of experts P_0 who make the right decision is $G_P M$; the expected value of the number of experts N_0 who make the wrong decision is $G_N M$; and the expected value of the number of experts that make no decision V_0 is $G_V M = \{1-(G_P+G_N)\}M$. At the first iteration of the reconciliation stage, the number of correct decisions will increase by $\frac{E_R G_P G_V}{G_P+G_N} M$, because the probability that correct decision is selected out of all decisions made is $\frac{G_P}{G_P+G_N}$; the probability that this selection is made by an expert who did not make a decision is G_V; and the probability that this decision is included in the group's decision is E_R. Therefore, the expected value of the number of correct decisions at the first iteration is $P_1 = G_P M + E_R G_P \frac{1-(G_P+G_N)}{G_P+G_N} M$. Let us make the same condition we used in the previous theorem, that $\frac{P_1}{M} > 0.5$ when $M \to \infty$, or after transformations, the condition of the theorem $E_R \frac{1-(G_P+G_N)}{G_P+G_N} > \frac{1-2G_P}{2G_P}$. Since when $M \to \infty$, the P_1 / M ratio is the probability that the expert makes correct decision at the first iteration of reconciliation stage, the condition of Condorcet's theory is met, and our theorem is therefore proven.

V. TECHNOLOGY FOR EVOLUTIONARY DECISION RECONCILIATION

There are many groups of experts that are not capable of making their first opinion correct with probability higher than *0.5*. Theorems proven above give us a hope for development of modern network programs that are able to overcome Condorcet's border and can be used efficiently in e-democracy systems.

The phenomenon when the probability that a group of experts makes correct decision increases due to the use of abilities of experts in selection of best decisions gave an opportunity to develop a new information technology for evolutionary decision reconciliation [9]. Multiple experiments in different creative fields confirmed the efficacy of the new approach. For example, collective intelligence was used to solve complex chess problems beyond the capabilities of individual group members; a group of witnesses effectively built a facial composite; a group of automated translators translated texts with higher quality than that of individual translations. IQ measurements using Eysenck verbal tests demonstrated group intelligence when the group was able to find correct answers to all 50 questions within a limited period of time [16]. One of the benefits of the new technology is that it gives an opportunity for objective measurements of individual expert contributions to the group project—both as idea generators and as evaluators of third-party decisions—and for development of hierarchical collective intelligence systems based on these measurements.

VI. CONCLUSION

The aforementioned results suggest that the evolutionary decision reconciliation technology is advisable for use in project management systems and e-democracy systems. New opportunities offered by this technology can expand the circle of potential contributors to collective solutions. Anonymous group effort, when experts work with ideas in text form without the need for personal contact with individuals with whom such contact is difficult due to psychological reasons, helps to fully unleash the intellectual potential of every expert. Use of genetic algorithms as group facilitators ensures quick convergence of the iterative process used to generate consolidated text. The technology based on evolutionary reconciliation method provides an opportunity to evaluate objectively the contribution of each expert to the consolidated product, and to establish a hierarchically organized self-governing crowdsourcing community[16]. Expert theorems proved in this work allow forecasting the probability that the decision will be correct as long as expert competence levels are known. The author and his colleagues are planning to continue research focused on practical use of this method to solve a variety of creative problems that expert communities are facing, and on the measurement of expert competence levels.

ACKNOWLEDGEMENTS

This work was supported by the Russian Foundation for Basic Research, grant #13-07-00958 “Development of the theory and experimental research of a new information technology of self-managed crowdsourcing”.

REFERENCES

[1] http://crowdsourcing.ru/sites/17, [retrieved: 04, 2014].

[2] Dutton W. H., “Networked Citizens and e-Democracy”, Conference on Citizenship and the Information Society, 1999, Lisbon, December 10, pp. 47-68 in Portugese, pp. 171-187 in English.

[3] http://andreo.li/blog/2-0-government-direct-democracy/, [retrieved: 05, 2014].

[4[http://www.njit.edu/, [retrieved: 05, 2014].

[5] J. Howe, “Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business”, Crown Business , 2008, p. 320.

[6] http://collectivescience.com/symintel.html, [retrieved: 05, 2014].

[7] P. Lévy, “Collective Intelligence: Mankind's Emerging World in Cyberspace”, 1994, p. 13.

[8] A. W. Woolley, C. F. Chabris, A. Pentland, N. Hashmi, T. W. Malone, “Evidence for a Collective Intelligence Factor in the Performance of Human Groups”, Science, 2010, V. 330 , pp. 686–688.

[9] V. Protasov, “Design of metasystem transitions”, Moscow, Physical and Technical Informatics Institute Press, 2009, p. 186.

[10] V. Protasov, Z. Potapova, “Self-Governing Crowdsourcing. Theory, Technology and Practice.”, Conference material of the 3rd International Conference on Control Automation, Intelligent Systems and Environments. Makhachkala, 2012, RAS Kabardino-Balkar Science Center Press, Vol. 1, pp. 91-99.

[11] V. F. Turchin., “The phenomenon of science a cybernetic approach to human evolution”, New York, Columbia University Press, 1977.

[12] V. Protasov, “Generation of new knowledge by network human-machine intelligence. Statement of the problem”, *J. Neurocomputers. Development and application*,, Moscow, 200,, vol.7-8, pp.94 – 103.

[13] V. Protasov, N. Vitiska, L. Shelhkova, “Use of collective intelligence of group of robots for acceleration of acceptance expedient decision”, *Conf.“Intellectual robotic system”,* Russia, Gelendjik, Oct. 2001, pp. 187— 189.

[14] V. Protasov, Z. Potapova, E. Melnikov, Overcoming the Condorcet's Border in Collective Intelligence Systems,“The Second International Conference on Intelligent Systems and Applications”, INTELLI 2013 April 21 - 26, 2013 - Venice, Italy, pp. 314-317.

[16] Le Marquis de Condorcet, “Essay on the Application of Analysis to the Probability of Majority Decisions”, 1785, Les Archives de la Revolution Française, Pergamon Press, pp. 9-23.

[16] V. Protasov, Z. Potapova, “Self-Governing Crowdsourcing. Theory, Technology and Practice. Conference material of the 3rd International Conference on Control Automation”, Intelligent Systems and Environments. Makhachkala, 2012, RAS Kabardino-Balkar Science Center Press, Vol. 1, pp. 91-99.

Dynamic Deployment and Reconfiguration of Intelligent Mobile Cloud Applications Using Context-driven Probabilistic Models

Nayyab Zia Naqvi, Davy Preuveneers and Yolande Berbers
iMinds-DistriNet, KU Leuven, Belgium
Email:{nayyab.naqvi, davy.preuveneers, yolande.berbers}@cs.kuleuven.be

***Abstract*—Today's mobile devices with advanced computing and storage resources are encouraging a whole new class of applications geared towards context-aware intelligence. However, for continuous processing of context-processing algorithms in interactive human-computing interfaces and augmented-reality experiences, these devices lack resource demands. Most of these intelligent applications rely on cloud paradigm for on demand computing, memory and storage resources. While combining both computing paradigms, finding the best strategy to deploy and configure intelligent applications, is not straightforward. In this paper, we analyse the challenges and requirements for the dynamic deployment of intelligent applications in such a federated setting. Additionally, a framework that leverages Dynamic Decision Network (DDN) is presented for decision making. DDN-based models deal with the presence of uncertainty and the partial observability of the context information, as well as the temporal effects of decisions to ascertain the Quality-of-Service and Quality-of-Context requirements. Our initial experiments with the framework demonstrate the feasibility of our approach and potential benefits to automatically make the best decision in the presence of a changing environment addressing the runtime variability.**

***Keywords*–dynamic deployment; probabilistic models; mobile cloud computing;intelligent applications.**

I. INTRODUCTION

With mobility and context-awareness [1], an ecosystem that can better connect the virtual world with the physical world is propagating. This propagation has multiplied the use of mobile devices in smart homes and offices, smart health, assisted living, smart cities and transportation. Research and development of context-aware intelligence aim at the ways to teach our devices about our environment narrowing the gap between us, our devices and the environment. Moreover, a relentless spurt of research activities in Mobile Cloud Computing (MCC) [2] aim to overcome the limitations of state-of-the-art mobile devices for intelligent applications. These cutting-edge applications require continuous processing and high-rate sensors' data to capture the users' context, such as their whereabouts and ongoing activities as well as the runtime execution context on the mobile devices. These devices use cloud resources to cope with the ever-growing computing and storage demands for its applications as a key enabler to run more demanding or long running tasks and applications.

The most viable technique in MCC is to dynamically adapt the mobile application by outsourcing the resource-intensive tasks to the cloud servers. Cloud computing can potentially save resources for mobile users [3][4]. Contrary to mobile devices, cloud computing provides plentiful storage and processing capabilities. This federated design could be applied to almost any application and has been shown to improve both the speed and energy consumption for non-trivial computations [5].

Nonetheless, not all applications are energy efficient when migrated to the cloud. Additionally, a lot of contextual information is sensed and captured on mobile devices. To deploy and initialize the desirable components automatically in this federated environment, involves several trade-offs with respect to the Quality-of-Service (QoS) [6] and the Quality-of-Context (QoC) [7]. A modular design philosophy for intelligent applications enables a more optimal deployment and performance when leveraging cloud technology. However, attaining the best deployment and configuration strategy for intelligent applications is not straightforward. In the core of such a distributed environment, adaptation decision of what to run where and when is non-trivial. The requirements of semantic knowledge and intelligence, the resource characteristics of the application components and low-level monitoring of the platforms used for deployment in terms of processing power, bandwidth, battery life and connectivity makes this decision highly dynamic. The main causes of this dynamism are runtime uncertainty and erratic nature of the context information [8], significantly impacting deployment decisions and performance. This context uncertainty can lead to annoying or incorrect decisions and can negatively impact the ability to reliably predict future contexts for proactive decision making as well. Furthermore, the system should be made aware of the impact of its decisions over time to optimize the runtime deployment and learn from its mistakes as humans do. The decision making needs to be flexible enough to ascertain the quality of its own decisions.

We present a framework designed to support the modular development and deployment of intelligent applications within the setting of MCC. We have adopted a probabilistic model based on DDNs for optimal decision making under evolving context of the runtime environment of a mobile device. DDN is an emerging research topic, and researchers are investigating its use in the area of self-adaptation for autonomous systems in several domains. Our major contribution in this paper is an approach of context-driven decision making using DDNs for dynamic deployment of intelligent applications, dealing with the dynamism and the partial observability of the context, as well as the temporal effects of the decision. Our model is able to learn deployment trade-offs of intelligent applications and capable of learning from earlier deployment or configuration mistakes to better adapt to the setting at hand. We have worked out our approach to an augmented-reality based use case where a DDN model decides and learns for the deployment of application components. Our experiments demonstrate the feasibility of the approach and potential benefits to automatically

 ISBN: 978-1-61208-352-0

make the best decision in the presence of changing situations or circumstances.

This paper is structured in six sections. In Section II, we give an overview of related work in MCC and discuss the gaps in state-of-the-art for federated deployments. Section III highlights the requirements and objectives of our use case scenario. Section IV provides a brief account of our proposed federated framework and the details about our approach of learning the trade-offs for dynamic deployments using a probabilistic decision model, mitigating the influence of runtime uncertainty. Finally, after evaluating our approach applied on our use case scenario in Section V, the paper concludes and offers a discussion of topics of interest for future work.

II. Background and related work

Although outsourcing the computation to the cloud can be beneficial in terms of QoS [6]. This distribution is never clear-cut in the scenario of intelligent systems considering trade-offs with respect to QoS and QoC [9]. The question is, where do you draw the line? What should be run on the cloud and what work should be done by the mobile device? The easiest option is to have everything stay local and not use the cloud or Internet at all, but as mentioned earlier, this could lead to bad performance of the application. Since the cloud consists of so many computers and has so much processing power, it is tempting to just decide to do almost all the work online. However, we can not forget that communication with the cloud has a price as well, economically and in terms of time and energy. In this case, multiple conflicting objectives affect the decisions.

Fortunately, researchers have developed a selection of systems that allow the application to make the partitioning decision while it is running, to provide the user with the best experience possible. CloneCloud [5] clones the entire set of data and applications from the smart phones to the cloud and selectively execute operations on the clone. However, for the continuous nature of context data, the CloneCloud [5] approach fails in a way that the dataset is not predetermined and needs to be processed in real-time for context-aware intelligence. Even more impressive, some of these platforms can let applications enjoy the best of computation offloading by only making minor changes, even when they have already been built without cloud computing capabilities [10]. Cloudlets [11] is another approach to achieve QoS but it does not provide an answer for the distribution of processing, storage, and networking capacity for each cloudlet. How to manage policies for cloudlet providers to maximize user experience while minimizing cost, security and trust are also open issues in order to adopt it for practical systems. Weblet [12] dynamically decides whether a weblet; a specialized form of HTTP-based web service interface, runs on the mobile device or the cloud with the help of Naive Bayesian learning techniques to find the optimal weblet configuration at runtime. Research is being done to investigate the usefulness of clouds accessing other clouds to reduce time and energy spent messaging back and forth between the mobile device and different cloud services [13].

Restating the obvious, intelligent applications have to take into account the runtime uncertainty in context data as context sources are dynamic in nature. They can disappear and re-appear at any time and context models change to include new context entities and types. The properties of context sources and context types can change randomly and the uncertainty can vary too. Fenton and Neil [14] have used Bayesian networks for predictions of the satisfaction of non-functional aspects of a system. Esfahani et al. [15] employ fuzzy mathematical models to tackle the inherent uncertainty in their GuideArch framework while making decisions on software architectures. Dynamic configuration of service oriented systems was investigated by Filieri et al. [16]. In contrast to our model, they used Markov models to investigate the decision making under uncertainty and quality requirements. Many works use utility functions to qualify and quantify the desirability of different adaptation alternatives. Bencomo and Belggoun [17] used DDNs to deal with the runtime uncertainty in self-adaptive systems. But these works are QoS-based, applied in different domains for resource allocation [18] and typically in component-based mobile and pervasive systems such as Odyssey [11] and QuA [19]. Markovian decision models still lack the tractability [20] for complex decisions under uncertainty.Moreover, the current state of the system has to be known, in order to use Markovian decision models. Several intelligent applications are pretty lightweight, but others require a lot of computational effort (e.g., for prediction) or require analysis of large amounts of data (e.g., for pattern analysis).

Our approach addresses the concern by identifying the deployment and performance trade-offs for outsourcing data and computation by continuously learning and adapting under multiple conflicting QoS and QoC objectives. If the decision is affirmative, required functionality is executed as a composition of loosely-coupled services on the cloud. Otherwise, lightweight component-based equivalents of these services are executed on the mobile.

III. Use case scenario

The use case considered in this paper is an AR mobile application called *Smart Lens*. Mobile Phones have been the most common platform for Augmented Reality (AR), ever since they were equipped with cameras. A mixture of AR and context-aware intelligence is often found in applications that aim to help the user explore certain places, be it cities, expos, museums or malls. In many cases, it makes the phone act as a camera but adds extra information next or onto objects that appear on screen. In short, mobile AR allows the devices to recognise objects as the user would, by seeing them.

A. Working procedure and requirements

By pointing to an electrical appliance, *Smart Lens* enables the users to be effortlessly aware of the power consumption of that appliance on top of that image. In addition to the near-real-time power consumption of the appliances, the *Smart Lens* application allows the user to get a detailed overview of the impact of any particular device compared to others in their electricity bill. Also, by storing the historic data it allows the user to monitor the performance of a device over time and determine whether it is time to replace a device with a more energy-efficient alternative. The modular design have been adopted for data and control processing on mobile and cloud ends to achieve a flexible distributed deployment. It simplifies redeployments and reconfigurations significantly.

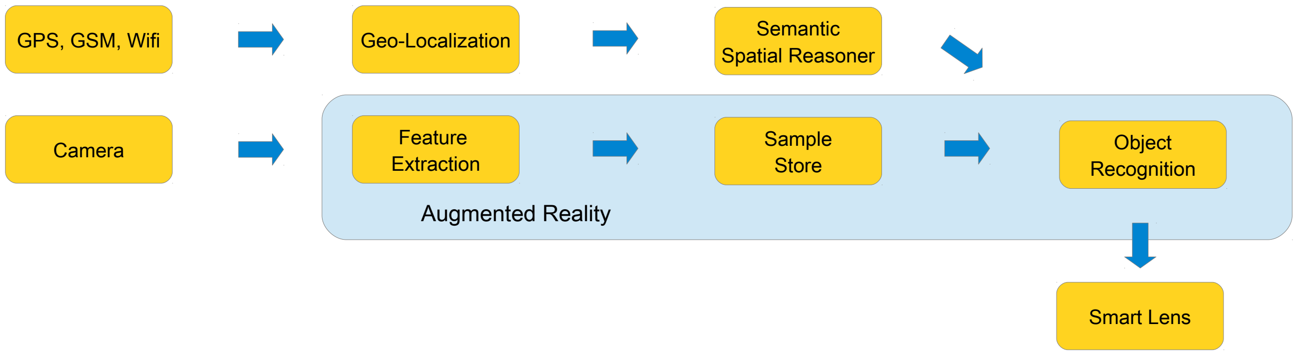


Figure 1. Deployment view with modular design philosophy for the components of *Smart Lens* application.

Figure 1, provides an overview of the composition of the components for our use case.

The major functional and non-functional system requirements for the use case are:

1) The system should be able to identify the electronic appliance, preferably with a markerless [21] approach.
2) The system should be able to capture the real-time context of the user and his environment.
3) The system must be adaptive at runtime to optimize the resource consumption of the application by outsourcing few components to the cloud.
4) The system should detect the runtime context of the mobile device i.e., CPU usage, memory consumption and battery usage. It should take into account the runtime evidence to mitigate contextual uncertainty and take the most rational decision.
5) The system should select the deployment strategy with respect to QoS & QoC requirements from the application's perspective, such as L_{min} for *Minimum Latency* and R_{max} for *Maximum Reliability* in terms of performance.

B. Objectives

Many opportunities for optimization exist as there are several distributed deployments of the application components and different configurations per component possible. The challenge is to find and analyse different optimization trade-offs in a federated environment of MCC, each characterized by varying sensing, communication, computation and storage capabilities. Moreover, mitigate the influence of runtime uncertainty in context and its quality.

A smart phone camera consumes its energy at a higher rate. The application might be able to identify an appliance but there are a lot of other appliances in a household, and user can point on them from any direction, so a conclusion based only on a single view of an appliance from one particular angle, is many times more likely to be incorrect. The *Semantic Spatial Reasoner* helps to maintain the reliability of the application. On top of that, *Feature Extraction* and *Object Recognition* techniques require a relatively large amount of processing time, which would not only drain battery further but causes the application and the phone as a whole to slow down as well.

The easiest option is to deploy every component locally on the mobile device and not use the cloud at all, but this could lead to bad performance of the application. Since the cloud consists of so many computers and has so much processing power, it is tempting to just decide to do almost all the work online. However, we can not forget that communication with the cloud has a price as well, economically and in terms of time and energy.

Based on the above mentioned requirements and shortcomings, we contrive the methodology for the well-informed decision making for dynamic deployment as follows:

1) A traditional QoS & QoC requirements gathering to identify and model the required quality attributes at design time. It is domain specific and involves the type of context being utilized.
2) Runtime support to detect a change in QoC in a particular context type and to measure its impact on other context types before making any decision.
3) Runtime support to detect a change in QoS before making any decision.
4) Enforcement of QoS & QoC policies or ways to ensure these requirements while making a decision.

In the next section, we will explain our context-driven dynamic deployment framework and its learning mechanism using probabilistic models to achieve well-informed decision making with the above described features.

IV. CONTEXT-DRIVEN DYNAMIC DEPLOYMENT APPROACH

Our framework consists of a loosely-coupled context-processing system, where each component has a dual implementation: an implementation for the mobile and a full-fledged scalable service equivalent running in the cloud. As shown in Figure 2, mobile hosts a *Dynamic Adaptation Module*, i.e., a client-side component of decision module to adapt the deployment configuration of the application. However, the cloud environment hosts a *Service Adaptation Module* which aims to optimize the runtime deployment of the required components and acts as an entry point for the adaptation module on mobile client. This module receives raw or pre-processed context data (including the type of the content and the identity of the source) and forwards it to a publish/subscribe subsystem so that interested parties (i.e., the subscribers) receive context updates.

 ISBN: 978-1-61208-352-0

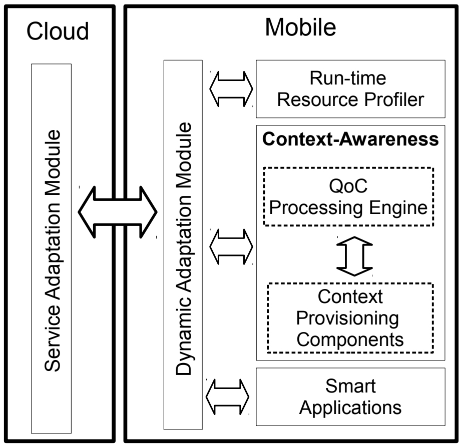


Figure 2. A blueprint of federated MCC framework and its dynamic deployment modules.

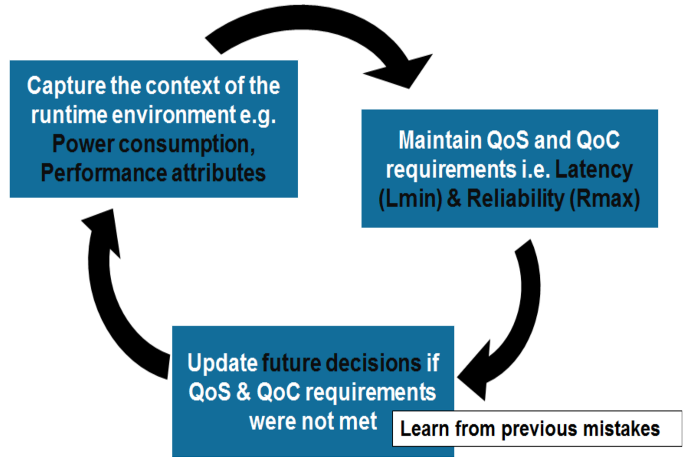


Figure 4. Enactment of dynamic decision making where system learns with it mistakes.

A. Deployment and reconfiguration decision making

Deployment Adaptation Module takes the decision of how to split responsibilities between mobile devices, applications and the framework itself. It should be able to decide for which use cases the cloud is better and for which ones a cloud based deployment does not bring any added value. Our decision making approach for redeployment and reconfiguration is explained in steps mentioned below:

Information discovery and selection – The framework discovers and explores application's runtime environment in order to get the context information to work with. Figure 3 shows a taxonomy of the runtime environment of a mobile device. Its resources can affect the QoS requirements and eventually, the decision of dynamic deployment.

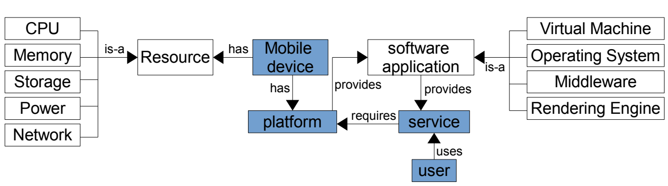


Figure 3. Taxonomy of the runtime environment of a mobile.

Our framework discovers the sensors and context types required for the smart application. Built-in sensors in mobile devices are important to fetch the context data, but the size of the acquired context data varies, depending on the application's objectives. The framework filters acquired context according to specific needs of the application. This selection process can be fairly complex as it may require complex filtering techniques to decide which sensor or device is offering relevant information. QoS and QoC requirements are gathered at this step to bootstrap the decision making. *Runtime Resource Profiler* component deployed on the mobile side, gathers these requirements.

Analysis and decision making – The framework uses discovered and selected information to make the deployment decision or change the configuration or behaviour of the application. For instance, user points his mobile device on an appliance. The image/video frame from the mobile device is captured. The feature points for the image in the frame are extracted. With these feature points , planar objects are then identified by matching the extracted feature points to those of known planar objects in a database. There is little memory because of his running video player. In this situation, a thin client configuration is chosen for *Smart Lens*, delegating *Spatial Reasoning* and *Object Recognition* components to cloud infrastructure. Furthermore, when the user shuts down video player, a context change is raised: free memory on the hand-held becomes high.

Enactment of the decisions – With a thin client configuration in previous step, the framework has to observe the real-time impact of the configuration to maintain QoS requirement of *Minimum Latency*. In order to increase application response time *Smart Lens* is reconfigured with a thick client and caching of data to save power and to become less vulnerable to network instability. As depicted in Figure 4, decision making is a continuous process, where the framework optimizes the application behaviour to reach certain objectives on the basis of the required attributes. When the availability of required resources varies significantly, framework has to decide whether to trigger an adaptation in the form of reconfiguration of the components. It learns from its previous decisions and the available context in order to ascertain the quality of the decision and learn from its own mistakes to achieve better results in future.

B. Learning the deployment trade-offs using DDNs

Conditional probability distributions derived by analysing historical attribute values helped solve stochastic problems in past. Runtime setting for every component is hard to determine in advance due to the dynamic interaction of these components with the environment and the user. Our *Deployment Adaptation Module* takes an advantage of probability theory and statistics to describe uncertain attributes. Probabilistic reasoning allows the system to reach rational decisions even when complete information is not available. Knowledge about runtime uncertainty can be captured by a data structure for probabilistic inference called a Bayesian network (BN). A BN is a Directed Acyclic Graph (DAG) represented by a triplet (N, E, P), where N is the set of chance nodes, E is the set of arcs to represent causal influence of the chance nodes and P is the conditional probability distribution for each chance node.

 ISBN: 978-1-61208-352-0

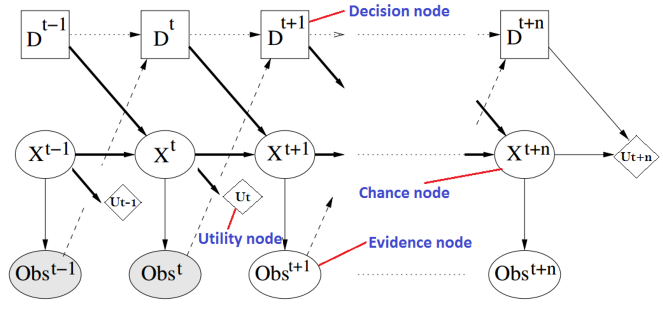


Figure 5. Structure of a DDN with dynamic chance nodes affecting utility nodes with decision nodes and evidences [22].

A Decision Network is a BN that also includes a set of decision and utility nodes. Utility nodes express the preferences among possible states of the world in terms of a subset of chance nodes and decision nodes. A probability-weighted expected utility is calculated for each decision given the evidence. To represent variables that change over time, it is possible to use a time-sliced network such that each time-slice corresponds to a time point. A DDN [17] is used for the enactment of the decisions that change over time influenced by dynamic states and preferences. To model the effectiveness of reconfiguration over time, decisions are modeled using a DDN where each time slice contains an action taken by the system. Figure 5 shows the structure of a general DDN.

DDNs are fed with runtime context of the user and the execution environment, such as CPU or memory usage, remaining battery and network connectivity. Chance nodes can represent runtime context, QoS or QoC requirements for each component of the intelligent application. Evidence nodes are different runtime situations based on the context of the user and parameters of the system. Decision nodes are the actual reconfiguration decision for each component. Utility nodes are influenced by chance nodes and decision nodes to infer the utility of dynamic deployment decision for a component. Utility functions can be used to assign priorities to different QoS and QoC requirements. The utility of redeployment decision is inferred using the QoS and QoC requirements of the application. System will choose the decision with highest expected utility, known as the Maximum Expected Utility (MEU) principle [22].

Our model expresses QoS_i, QoC_i and the context of the mobile device by chance nodes. These chance nodes make a BN with conditional probabilities corresponding to the effects of different actions D_j over QoS_i or QoC_i. Evidence nodes, defined as "Obs" in Figure 5, express the uncertainty factors connected to the chance nodes to take a favorable decision. For each dynamic chance node, utility node expresses the utility function that takes conditional probabilities of QoS or QoC requirements and their priorities into account. The expected utility for each decision is computed with respect to the $P(QoS_i|QoC_i,D_j)$, $P(QoC_i|D_j)$ and a weight for the decision as expressed in Equation (1).

$$EU(D_j \mid QoC_i) = {}_i\Sigma P(QoS_i \mid QoC_i, D_j)U(QoS_i \mid D_j) \quad (1)$$

V. EXPERIMENTAL EVALUATION

In order to implement our framework, discussed in Section IV-A, we are using an HTC One X with a 1.5 GHz Quad Core ARM Cortex processor (at about 2.5 MIPS per MHz per core) to run the Android-based *Smart Lens*, augmented reality application, which embeds the Vuforia library to recognize everyday devices of the users in the environment. Our infrastructure runs VMware's open source Platform-as-a-Service (PaaS) offering known as Cloud Foundry on a server with 8 GB of memory and an Intel i5-2400 3.1 GHz running a 64-bit edition of Ubuntu Linux 12.04. A Java-based implementation has been used for *Runtime Resource Profiler* that captures the runtime context of mobile device. We have modeled a DDN-based probabilistic model for the components of our *Smart Lens* use case using JSmile Android API in Genie and Smile environment from Decision Systems Laboratory [23]. Our DDN model is evaluated using Equation (1) for every decision D_j computing the probability-weighted average utility for that decision.

The initial experiments are conducted for the redeployment of *Object Identification* component. Our model has been designed and expanded for 5 time slices as shown in Figure 6. CPU and memory usage are modeled as static chance nodes representing execution context. QoS requirement *Minimum Latency* is modeled as a dynamic chance node influencing the redeployment decision and QoC requirement *Maximum Reliability* is observed in terms of performance. The QoS and QoC requirements both affect utility node. Decision node is the actual reconfiguration decision for each component.

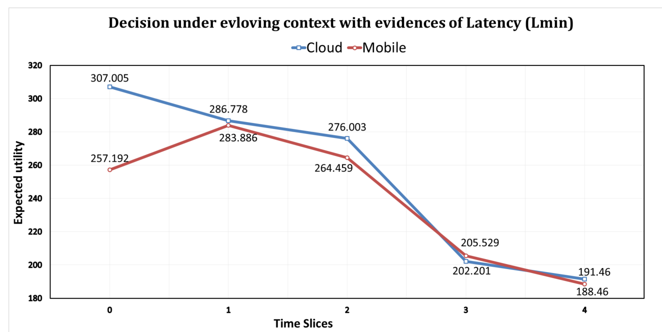


Figure 6. Expected utility for redeployment decision of *Object Identification* component with changing runtime situations.

QoS requirement of latency is kept minimum with respect to a threshold value of 3 seconds. Initial conditional probabilities are defined with respect to QoC requirements where the CPU usage is kept less than 40% and memory usage less than 45 MB. Figure 6 shows that our model learns and adapts according to a change in the context of the runtime environment at each time slice, where each time slice is realized by a ticker initiated from *Runtime Resource Profiler*, whenever the runtime environment changes. At time slice 4, latency is observed and model adapts itself to deploy this component on the mobile device rather than the cloud, according to the required QoS.

VI. CONCLUSION AND FUTURE WORK

The optimal strategy to deploy and configure intelligent applications with dynamic and heterogeneous resource availability is not forthright. This deployment of software components has to take into account the resource characteristics of application components and the platforms used for deployment in terms of processing power, bandwidth, battery life and connectivity. Our modular design philosophy for developing intelligent applications helps to dynamically configure, compose and deploy these components. The overall aim of our work is to intelligently automate the distributed deployment and configuration of the components across the mobile and cloud infrastructures.

In this paper, we have presented a novel approach for dynamic deployment decision making in federated environment of MCC by leveraging DDN to automate decisions in a continuously evolving runtime environment context. DDNs build upon Dynamic Bayesian Networks. However, the latter is only able to learn conditional probabilities based on a dataset, whereas DDNs can quantify the impact of the evidence and the effect of the decisions. Furthermore, by exploiting the utility of deployment decisions, our framework can learn how to automatically improve its decisions in the next iteration or time slice. Our first contribution was a feasibility analysis of incorporating DDNs for decision making, and our experiments have clearly demonstrated the ability of adapting its decision in the presence of evolving situations and an uncertain context of the environment. By incorporating QoS & QoC in our DDNs, we are able to assess the quality of our context-driven decisions, ascertain their quality and update future decisions and corresponding actions according to the outcome and impact. Our preliminary experiments have shown that an intelligent application can achieve optimal deployment for its components, whenever the context is updated. However, the sensitivity analysis in federated environment has to be conducted.

Further work is required towards more systematic techniques for the runtime synchronization of multiple DDN models and to empirically study the scalability of these models. The value of the probabilities that change over time and their impact on alternative decisions can also be of interest. Finally, developing tools to specify the QoC requirements and design a DDN would be certainly very helpful as current tool's support for modelling and using DDNs is fairly limited.

ACKNOWLEDGMENT

This research is partially funded by the Inter University Attraction Poles Programme Belgian State, Belgian Science Policy, and by the Research Fund KU Leuven.

REFERENCES

[1] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *Handheld and ubiquitous computing*, pp. 304–307, Springer, Springer, 1999. [retrieved: March, 2014].

[2] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, pp. 1587–1611, 2013. [retrieved: January, 2014].

[3] K. Kumar and Y.-H. Lu, "Cloud computing for mobile users: Can offloading computation save energy?," *Computer*, vol. 43, no. 4, pp. 51–56, 2010.

[4] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, pp. 4–4, 2010.

[5] B.-G. Chun and P. Maniatis, "Augmented smartphone applications through clone cloud execution.," in *HotOS*, vol. 9, pp. 8–11, 2009. [retrieved: Feb, 2014].

[6] D. Chalmers and M. Sloman, "A survey of quality of service in mobile computing environments," *Communications Surveys & Tutorials, IEEE*, vol. 2, no. 2, pp. 2–10, 1999. [retrieved: October, 2013].

[7] T. Buchholz, A. Küpper, and M. Schiffers, "Quality of context: What it is and why we need it," in *Proceedings of the workshop of the HP OpenView University Association*, vol. 2003, Geneva, Switzerland, 2003. [retrieved: June, 2013].

[8] N. Chen and A. Chen, "Integrating context-aware computing in decision support system," in *Proceedings of the International MultiConference of Engineers and computer Scientists*, vol. 1, pp. 359–364, 2010. [retrieved: June, 2013].

[9] N. Z. Naqvi, D. Preuveneers, Y. Berbers, *et al.*, "Walking in the clouds: deployment and performance trade-offs of smart mobile applications for intelligent environments," in *Intelligent Environments (IE), 2013 9th International Conference on*, pp. 212–219, IEEE, 2013. [retrieved: January, 2014].

[10] Y. Zhang, G. Huang, X. Liu, W. Zhang, H. Mei, and S. Yang, "Refactoring android java code for on-demand computation offloading," in *Proceedings of the ACM international conference on Object oriented programming systems languages and applications*, pp. 233–248, ACM, 2012.

[11] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *Pervasive Computing, IEEE*, vol. 8, no. 4, pp. 14–23, 2009. [retrieved: December, 2013].

[12] X. Zhang, A. Kunjithapatham, S. Jeong, and S. Gibbs, "Towards an elastic application model for augmenting the computing capabilities of mobile devices with cloud computing," *Mobile Networks and Applications*, vol. 16, no. 3, pp. 270–284, 2011. [retrieved: December, 2013].

[13] S. Kosta, V. C. Perta, J. Stefa, P. Hui, and A. Mei, "Clone2clone (c2c): Peer-to-peer networking of smartphones on the cloud," in *5th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud13)*, 2013. [retrieved: February, 2014].

[14] N. Fenton and M. Neil, "Making decisions: using bayesian nets and mcda," *Knowledge-Based Systems*, vol. 14, no. 7, pp. 307–325, 2001. [retrieved: December, 2013].

[15] N. Esfahani, K. Razavi, and S. Malek, "Dealing with uncertainty in early software architecture," in *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering*, ACM, 2012. [retrieved: January, 2014].

[16] A. Filieri, C. Ghezzi, and G. Tamburrelli, "A formal approach to adaptive software: continuous assurance of non-functional requirements," *Formal Aspects of Computing*, vol. 24, no. 2, pp. 163–186, 2012. [retrieved: December, 2013].

[17] N. Bencomo and A. Belaggoun, "Supporting decision-making for self-adaptive systems: from goal models to dynamic decision networks," in *Requirements Engineering: Foundation for Software Quality*, pp. 221–236, Springer, 2013. [retrieved: June, 2013].

[18] T. Kelly, "Utility-directed allocation," in *First Workshop on Algorithms and Architectures for Self-Managing Systems*, vol. 20, 2003. [retrieved: January, 2014].

[19] S. Amundsen, K. Lund, F. Eliassen, and R. Staehli, "Qua: platform-managed qos for component architectures," in *Proceedings from Norwegian Informatics Conference (NIK)*, pp. 55–66, 2004. [retrieved: March, 2014].

[20] P. C. Da Costa and D. M. Buede, "Dynamic decision making: a comparison of approaches," *Journal of Multi-Criteria Decision Analysis*, vol. 9, no. 6, pp. 243–262, 2000.

[21] E. Ziegler, "Real-time markerless tracking of objects on mobile devices," 2012. [retrieved: December, 2014].

[22] S. J. Russell, P. Norvig, J. F. Canny, J. M. Malik, and D. D. Edwards, *Artificial intelligence: a modern approach*, vol. 2. Prentice hall Englewood Cliffs, 1995. ISBN: 0131038052.

[23] M. J. Druzdzel, "Smile: Structural modeling, inference, and learning engine and genie: a development environment for graphical decision-theoretic models," in *AAAI/IAAI*, pp. 902–903, 1999. [retrieved: March, 2014].

A Method of Applying Component-Based Software Technologies to Model Driven Development

Keinosuke Matsumoto, Tomoki Mizuno, and Naoki Mori
Dept. of Computer Science and Intelligent Systems
Graduate School of Engineering, Osaka Prefecture University
Sakai, Osaka, Japan
matsu@cs.osakafu-u.ac.jp

***Abstract*—Improving the efficiency of automatic software generation is a problem of model driven development method. In order to solve this problem, we apply component-based software technologies that have been mainly developed on software implementing level to modeling level. In the proposed method, functionally relevant model elements are packed as a model component, and modeling software is carried out by associating it with each model component. The role of model components becomes clear by introducing the concept of a component, and the reuse of model components rises by externalizing the dependency between components. In addition, flexible model transformation rules linked to the role of model components can be designed. As a result, it is possible to automatically generate more source code. The validity of the proposed method has been proved by application experiments.**

Keywords-model; component-based development; object oriented design; UML; MDA

I. Introduction

Model driven development (MDD) method which is based on model driven architecture (MDA) [1] draws attention as a technique that can flexibly deal with changes of business logics or implementing technologies in the field of system development. Its core data are models that serve as design diagrams of software. It includes a transformation to various kinds of models and an automatic source code generation from the models. These models themselves are expected to be reused [2]. By reusing models, you can also reuse the knowledge that does not depend on platforms. It can improve software development efficiency. For that purpose, it is necessary to pack the highly relevant model elements into a function and to clarify a reuse unit [3].

On the programming level, there is a development technique called component-based software method [4] that especially aims at reusing the source code. A component is a set of highly relevant and reusable program parts. The component-based method develops software by combining components. It has been mainly improved in the programming field, and various development frameworks are now put into practice. In recent years, component-based modeling [5] which applies the concept of components developed on programming to software models has been advocated. The software models treated by this component-based modeling and the models treated by MDD are the same.

You can develop the component-based modeling further and apply it to MDD. According to this idea, this paper proposes a method that makes model elements loose coupling by introducing a component of the component-based technologies to MDD. It also describes a technique to design automatic generation rules that is one of the features of MDD. Finally, this study aims at increasing the efficiency of MDD by the proposed software development method.

The structure of this paper is shown below: Section II describes component-based software technologies, and Section III explains the proposed method of this research. In Section IV, we show the results of application experiments in order to confirm the validity of the proposed method. Finally, Section V describes conclusions and future subjects.

II. Component-based Software Technologies

Fundamentally, a component is not used alone, but it is used in order to build software by combining it with other components or programs. This study uses software patterns called Inversion of Control (IoC) and Dependency Injection (DI).

A. Inversion of Control

IoC is a kind of software architecture, which frameworks or containers actively call components [6]. On the contrary, components do not call other cooperative components, but they only have references to their interface instead. Components do not need to know other cooperative components at the time of software implementation. This promotes interface programming and stimulates loose couplings of software.

B. Dependency Injection

By realizing IoC through interfaces, you need a mechanism to get the required components at the time of execution. DI [7] is a technique to realize the mechanism. Dependency is injected outside components. The dependency is injected outside the components and described apart from the source code expressing business logics. A container managing the life cycle of components analyzes the dependency at the time of execution and specifies the components to cooperate with according to the dependency. It is not necessary to describe the process of acquiring external components in the source code which realizes business logics. Therefore, IoC is realized.

III. PROPOSED METHOD

Dependency between components is externalized by introducing IoC into a model level, and loose coupling between components can be also promoted. The proposed method adopts this approach in MDD. It aims at improving the efficiency of MDD by increasing the efficiency of modeling, reusing models, and defining an externalized dependency between components. This section explains the modeling process suggested in the proposed method by using Unified Modeling Language (UML) diagrams. The position of the proposed method is shown in Fig. 1. Fig. 2 shows a flow of development cycle in the proposed method.

A. Modeling

Modeling is an approach for introducing IoC in class relations, and messages passing through components and container interfaces. This approach is close to the Catalysis approach [3], which decides on component interfaces in advance and models the inside of the components independently. However, the proposed method is not necessary to know external components because it adopts the IoC, and it differs from the Catalysis approach in this point. The proposed method condenses software modules that are functionally connected as a model component. It is required to decide component granularity by introducing the concept of components into upstream processes from the implementing level. Components are classified by the granularity of practical software examples as follows:

1) Business Component: It corresponds to one of business processes like an order receipt, estimate. It is realized by compounding business functional components.

2) Business Functional Component: It is a business element unit like accepting order receipt, replying estimation request. A service in Service Oriented Architecture (SOA) is located here.

3) System Functional Component: It is a system part like registration, updating and reference of information. A business functional component is created using the system functional components.

The proposed method targets at the system functional components among various granularities.

B. Externalizing Dependency of Model Elements

The proposed method also designs a model for defining dependency between model components. This is referred to as an external cooperation model. A component diagram of UML is used for modeling of external cooperation models. The component diagram, which shows a static structure of software, can be divided into demand interfaces and supply interfaces. It can describe an interface and the dependency, including its directivity [8]. The proposed method calls an internal structure model that shows the inside of the model component structure. To design the internal structure model, we use a class diagram of UML. However, the class diagram makes use of internal classes, interfaces, libraries, and demand/supply interfaces that belong to the component. It does not directly use external elements to keep IoC. The external cooperative models are defined outside of the

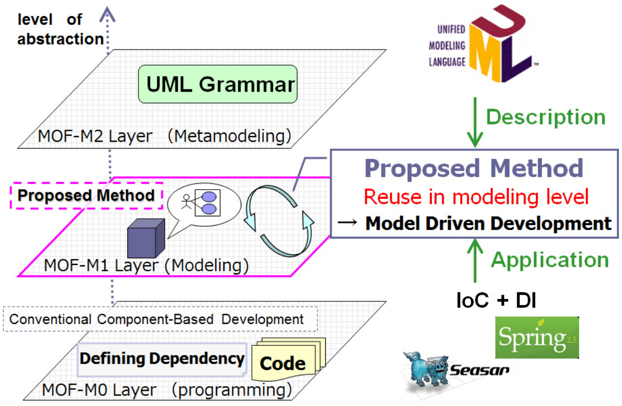


Figure 1. Position of the proposed method.

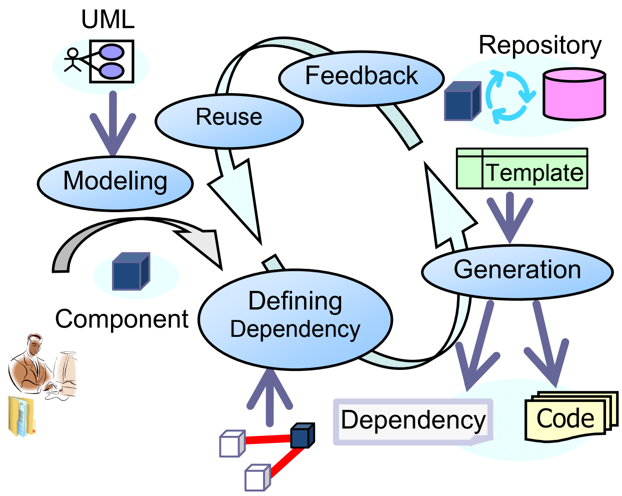


Figure 2. Cycle of software development.

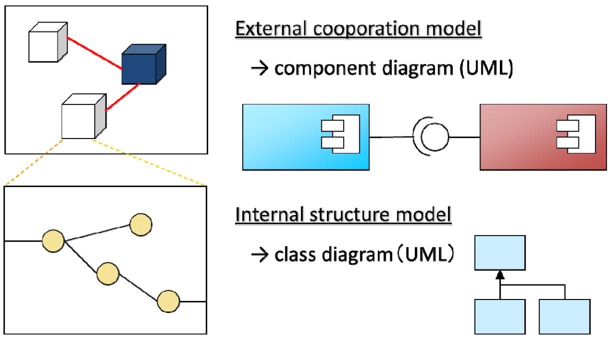


Figure 3. Dependency of model elements.

components. The concept of IoC is introduced into a model level, and externalizing dependency between components is achieved. Fig. 3 shows these concepts.

C. Selective Design of Model Transformation Rules

In order to transform models to texts like source code, you need to design a template describing transformation rules from the model to the texts. The proposed method makes the role of components clear by introducing the concept of a component and it enables to design a template dedicated to a component. For example, prominent software patterns like analysis patterns [2] and design patterns [9] [10] [11] can be used and employed as model components or transformation rules. The proposed method does not limit the type of the system for development. It is advantageous that a developer can give role information freely to model components or transformation rules.

It is very difficult to design a general-purpose template in a platform. If a template is designed for a specific project [12], it will become hard to divert it to other projects. The proposed method designs a template for each component because it is separated according to a function of the component. For this reason, the template design becomes more flexible than ever. It makes use of the externalizing dependency and no need to write special code for DI in source code.

IV. Application Experiments

The proposed method is applied to a sample system to confirm the effectiveness of the proposed method.

A. Experimental Overview

The sample system is a typical three layer client server type of Web application. The main roles, implementing methods, and applied design patterns are shown in the following:

1) Business Logic Layer: It corresponds to functional requirements of the Web application. It receives inputs from a controller and sends processing results to a presentation layer. In addition to service components, a naming service by Java Naming and Directory Interface (JNDI) is realized in order to make possible Remote Method Invocation (RMI) of procedure. Service Locator, Business Delegate, Singleton, and Proxy are used as design patterns.

2) Data Access Layer: It abstracts a persistent data system, such as a database or a file system. It is a uniform window to deal with the data system. Data Access Object, Factory Method, Abstract Factory, and Facade are used as design patterns.

3) Entity: It is a class group expressing the data that serve as persistent objects among the object classes in a problem domain. Data Transfer Object (Value Object), Composite Value Object, and Value Object Assembler are used. In addition, platform specifications are shown in the following:

- Programming Language: Java Development Kit 5.0.
- Application Server: Glassfish Open Source Edition v3.
- Database Management System: MySQL 5.1

Many MDA tools generate skeleton code from class diagrams of UML. For comparison, these tools are regarded as conventional methods. A standard sample of Acceleo [13] is used in this experiment. The Acceleo sample has an automatic generation function of Java source code from class diagrams designed by UML modeling tool UML2 of Eclipse. We compare three kinds of experimental data: the source code automatically generated by the proposal method, the source code generated by the conventional method, and the final source code. The last one is composed of the source code resulting from the proposed method and hand coded additional source code. These experimental data are respectively called "proposed method", "conventional method", and "finished goods".

B. Template Design

Fig. 4 is an example of the dedicated template that generates the code of a class method, implemented by Java+Spring Framework, for Web page controller models. This model component has a Web page controller profile, as confirmed by the lines 1-2. An appropriate function is given to a class method by detecting the applied stereotype as shown in the lines 3-9. This template is stored in the model component of the Web page controller. The proposed method enables to design a specialized template in a UML profile. The template is also united with a role, but it is not based on a field. In addition, the automatic generation rate of code also becomes higher.

The templates are prepared by roles, such as a template corresponding to design patterns, controllers in a client server type, and database access. A template reads information on a model and is a medium that generates source code. The automatic generation is carried out by each model component. The source code is merged with information on the internal structure model, the external cooperation model, and the UML profile.

C. Evaluation by Abstract Syntax Tree

This experiment measures the number of Abstract Syntax Tree (AST) nodes in order to evaluate the quantity of the source code generated automatically according to procedure of the MDD. The AST syntactically analyzes the source code and expresses it in a directed tree. There are two advantages of investigating AST nodes: The first one is that it may be easier to reflect the actual processing of a program than with conventional indexes, such as Line of Code (LOC): number of source code lines. The second one is that it can investigate a type of nodes that constitutes the AST. This information is useful when knowing the structure of a program. On the contrary, LOC is a simple numerical value and cannot know the structure of the program.

This experiment uses Eclipse Java Development Tools (JDT) as an API that builds AST from Java source code. In addition, language specification is AST-Java Language Specification3 (JLS3). JLS3 is equivalent to Java Development Kit 5.0. When LOC is calculated, you may

```
[if (not o.eContainer(Package).getAllAppliedProfiles()
    ->select(p : Profile | p.qualifiedName= 'test.web.controller')->isEmpty())]
[for (s : Stereotype |o.getAppliedStereotypes())]
[if (s.name = 'GET')]
    @RequestMapping(method = RequestMethod.GET)
[elseif (s.name = 'POST')]
    @RequestMapping(method = RequestMethod.POST)
[/if]
[/for]
[/if]
    public [o.returnTypeOperation()/]([o.getInParameter()/]) {
[if (not o.type.oclIsUndefined())]
        // [protected ('for operation '.concat(o.name))]
        // TODO should be implemented
        return null;
        // [/protected]
[else]
        // [protected ('for operation '.concat(o.name))]
        // TODO should be implemented
        // [/protected]
[/if]
    }
```

Figure 4. Example of role-specific template for Acceleo.

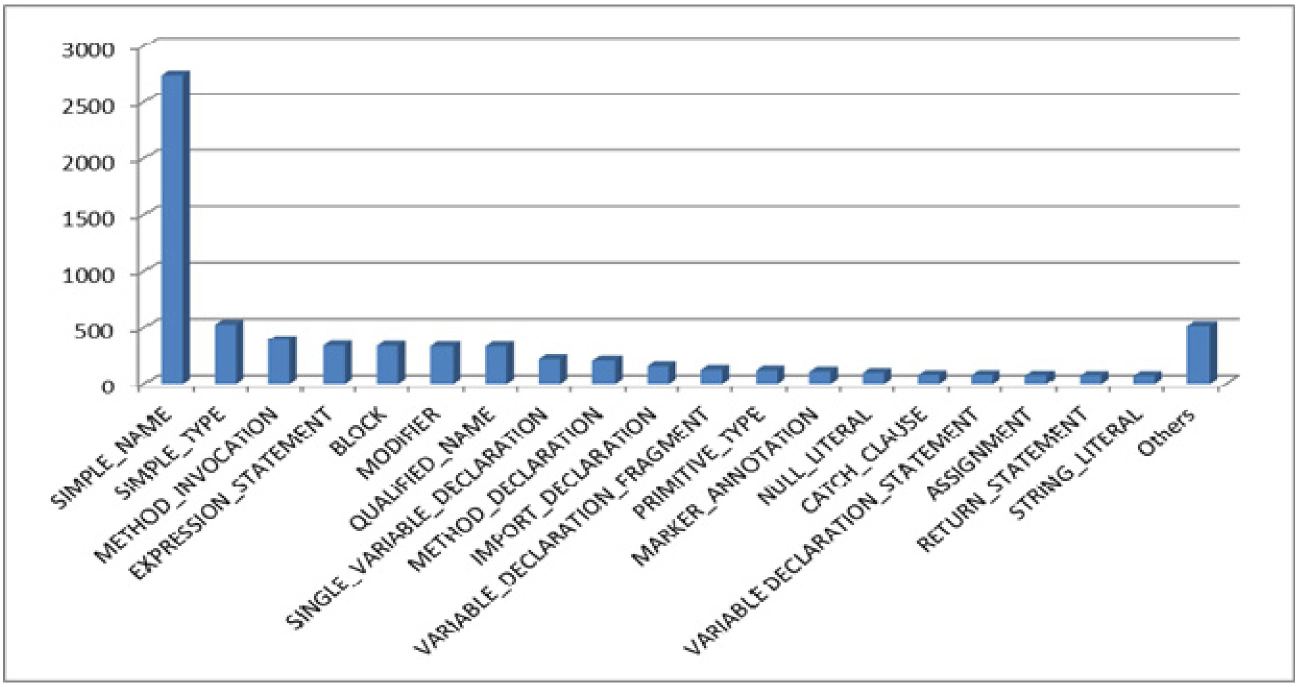


Figure 5. Breakdown of AST nodes generated by the proposed method.

disregard lines that are unrelated to the program execution such as comment lines. The value measured on the condition is called a logical LOC. Like this, this experiment eliminates LINE COMMENT, BLOCK COMMENT, and JAVADOC that express comments when you add up AST nodes.

Fig 5 indicates each AST node's category of the proposed method and the number of nodes belonging to the category. All graphical models that serve as inputs of the automatic generation tools are the same. The total numbers of AST nodes for the finished goods, the proposed method, and the conventional method are 7307, 6859, 2200, respectively. The execution time of the conventional method is really shorter than the one of the proposed method. However it could generate the third of AST nodes of the

 ISBN: 978-1-61208-352-0

proposed method, and includes many errors in the generated source code.

D. Discussion

The number of AST nodes is remarkably increased by the proposed method in comparison with the conventional method. It turns out that class methods occupy a large proportion in the source code of a system. The proposed method is effective in MDD from this point. In addition, generated AST nodes of the proposed method are investigated in detail as compared with the conventional method. SIMPLE NAME nodes are ranked number one. They express class names, field names, class method names, variable names, and so on, as it appears everywhere in the source code. They are located in terminal nodes in AST. Therefore, there are more SIMPLE NAME nodes than other nodes in both the proposed method and the conventional method. From this point of view, it cannot show advantages of the proposed method. The other AST nodes that appear in class methods are investigated except the SIMPLE NAME nodes. The total of the increasing amount of these AST nodes, except for SIMPLE NAME nodes, is 1508 and it occupies about 32.4% of the whole increasing amount 4659. It becomes about 56.7% if the SIMPLE NAME nodes are eliminated. Fig. 6 is a bar graph in which incremental values of automatic generated code of the proposed method are ranked in descending order in comparison with the conventional method by the category of AST nodes. However, this graph omits the SIMPLE NAME nodes and the nodes which incremental amount are zero.

Regarding applied design patterns, it is easy to design templates for Service Locator, Business Delegate, and Data AccessObject. These patterns are more commonly used to describe the bodies of class methods. The proposed method is superior in dealing with design patterns. The readability of the generated source code is expected to be high. The readability is one of the elements that is not described in models but appears in source code. It is easy to describe it as

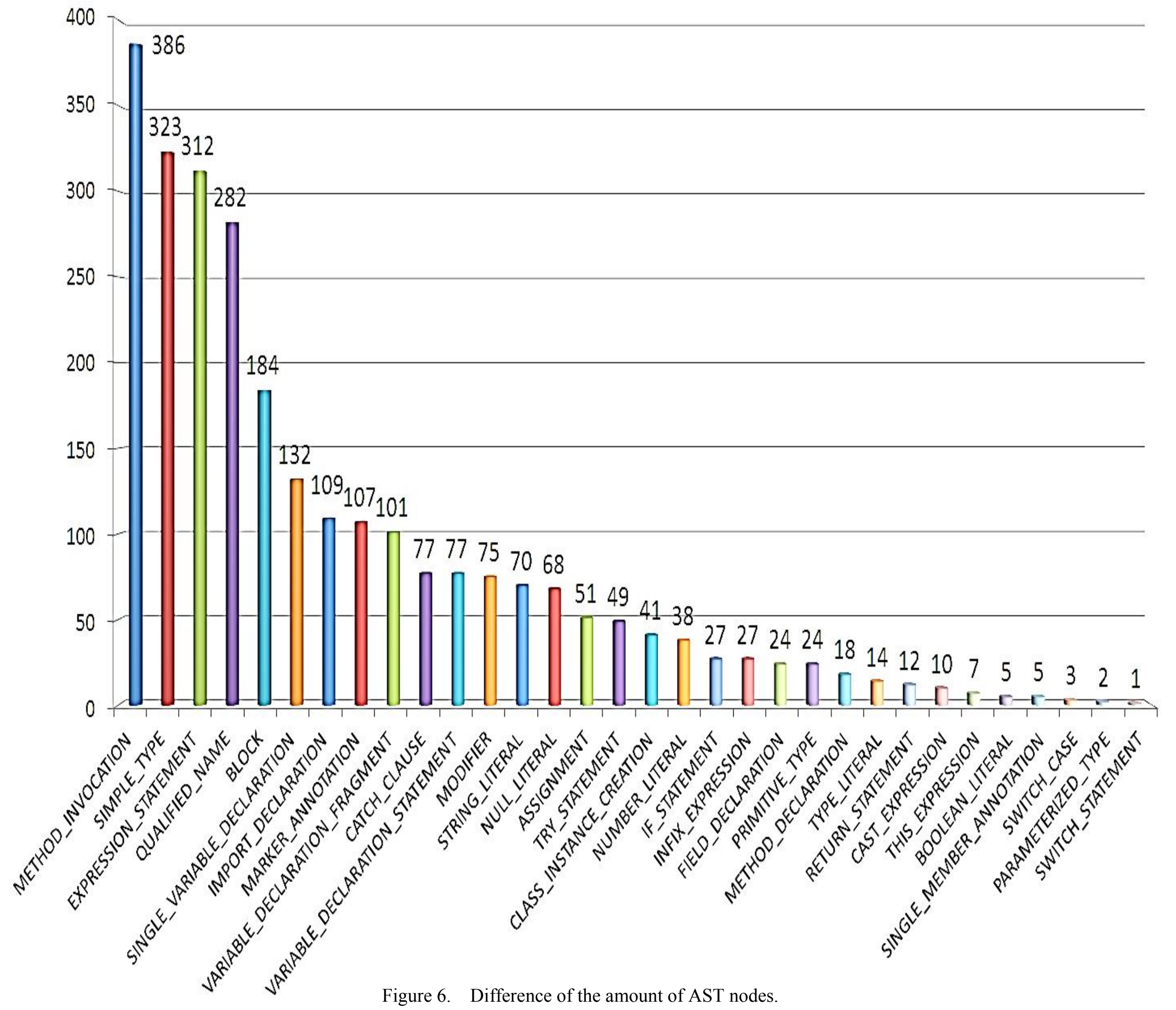


Figure 6. Difference of the amount of AST nodes.

coding conventions in templates. Although it is necessary to design templates for every role of the model components by the proposed method, they are independent from fields and they are reusable for other projects.

AndroMDA [14], Software Factories [15], and UML Components [16] do not give class methods. They are almost the same as with the conventional method. BridgePoint [17] that gives class methods by an action description language makes a little difference with the proposed method about automatic generation rates of source code. But it does not develop certain patterns using the concept of IoC and DI. It is hard to design templates and to build them individually. Some corrections are needed to reuse the component models and templates for other examples, and it has a low chance of reuse either.

V. Conclusions

This paper has proposed a software development method that applies component-based technologies to MDD. The proposed method was compared with the conventional method using a sample Web system. This method is able to automatically generate more than three times the amount of AST nodes compared with the conventional method. The proposed method can describe information concerning class methods in the templates. Especially, describing functions of source code or main parts of methods in templates improves the rate of automatic code generation. This fact shows the advantages of the proposed method concerning the generation efficiency of class methods, by designing templates linked to the role of model components.

On the other hand, insufficient parts of the proposed method as compared with the finished goods are mainly business logics. They are class methods that belong to business components. It is difficult to build templates for these parts because they are easily affected by requirement specifications. As a result, the automatic generation rate is low. As a future subject, the proposed method should deal with a different granularity, like business components. In addition, a further extension could be to take into account dynamic behavior diagrams.

Acknowledgment

This work was supported in part by JSPS KAKENHI Grant Number 24560501.

References

[1] S.J. Mellor, K. Scott, A. Uhl, and D. Wiese, MDA Distilled, Addison Wesley Longman Publishing Co., Inc. Redwood City, CA, 2004.

[2] M. Fowler, Analysis Patterns: Reusable Object Models, Addison-Wesley, 1997.

[3] D.F. D'souza and A.C. Wills, Objects, Components, and Frameworks with UML: The Catalysis Approach, lavoisier.fr, 1998.

[4] I. Crnkovic, “Component-based software engineering - new challenges in software development,” Software Focus, Vol.2, Dec. 2001, pp. 127-133, doi:10.1002/swf.45.

[5] G. Gossler and J. Sifakis, “Composition for component-based modeling,” Science of Computer Programming, Vol.55, Mar. 2005, pp. 161-183.

[6] D. Rosenburg and M. Stephens, Use Case Driven Object Modeling with UML: Theory and Practice, Apress, New York, 2007.

[7] Spring framework, http://www.springsource.org/, [retrieved: April, 2014].

[8] Object Management Group, OMG unified modeling language superstructure specification, V2.1.2, 2007.

[9] E. Gamma, R. Helm, R. Jhonson, and J. Vlissides, Design Patterns: Eelements of Reusable Object-Oriented Software, Addison-Wesley, 1995.

[10] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal, Pattern-Oriented Software Architecture: A System of Patterns. Wiley, 1996.

[11] D. Alur, D. Malks, and J. Crupi, Core J2EE Patterns: Best Practices and Design Strategies, Prentice Hall, 2001.

[12] K. Matsumoto, T. Maruo, M. Murakami, and N. Mori, “A graphical development method for multiagent simulators,” in Modeling, Simulation and Optimization: Focus on Applications, S. Cakaj, Ed. Vukovar: In-Tech, 2010, pp. 147-157.

[13] Acceleo, http://www.eclipse.org/acceleo/, [retrieved: April, 2014].

[14] AndroMDA, http://www.andromda.org/, [retrieved: April, 2014].

[15] J. Greenfield and K. Short, “Software factories: assembling applications with patterns, models, frameworks and tools,” Proc. Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, Oct. 2003 pp. 16-27.

[16] J Cheesman and J Daniels, UML Components. Addison-Wesley, 2001.

[17] BridgePoint, http://www.bridgepoint.eu/, [retrieved: April, 2014].

ℜeALIS1.1: The Toolbox of Generalized Intensional Truth Evaluation

Gábor Alberti and László Nőthig
University of Pécs, Department of Linguistics
ℜeALIS Theoretical, Computational and Cognitive Research Team, Pécs, Hungary
emails: {alberti.gabor@pte.hu, nothig.laszlo@gmail.com}

***Abstract*—Our software ℜeALIS1.1 is primarily intended to supply linguists with a highly intelligent device to build fragments of languages. Then (non-linguist) experts can elaborate a peculiarly "multiplied" database that offers, besides the model of the external world, hundreds of its (appropriately labeled) alternatives. According to the ℜeALIS theoretical framework we use, these alternative models can all be linked to human agents represented in the world model as their pieces of knowledge, beliefs, desires, intentions, dreams. Finally, (further) users can select lexical items to build sentences, the (generalized) truth-conditional interpretation of which they will be given on the basis of the actual version of the above-sketched "multiplied world model". Our software serves not only the theoretical purpose of trying out the "pragma-linguistics" theory (by implementing it), but also the practical purpose of collecting and systematizing data in the peculiar structure due to ℜeALIS, intended to truly capture human intelligence.**

Keywords—dynamic discourse semantics; possible worlds; truth-conditional interpretation; presupposition.

I. INTRODUCTION

We are working on the implementation of a pragma-linguistics theory, ℜeALIS (*Reciprocal and Lifelong Interpretation System*), intended to truly capture human intelligence by means of a peculiarly multiplied world model. The implementation of this "intelligent" structure is our primary innovation.

At the moment, it is very important to us that the software, which is permanently developed, has repercussions on the theory, due to the fact that the theory can be tested and sophisticated by means of the software. In a later period the practical purpose of collecting and systematizing data in the peculiar structure due to ℜeALIS will stand out.

The main difficulties lie with the fact that we are working on a completely general toolbox that utilizes the afore-mentioned multiplied world model, that is, one which is underspecified in several respects but can be rapidly specified when it is designated for particular purposes. This also holds for the linguistic input; some difficult grammatical phenomena that should be captured in a demanding way are collected in Section VI. Due to the uniform and holistic approach of ℜeALIS, we cannot afford to use parsers or other devices developed in other projects.

Recursivity is another stubborn problem. Unlimited chains of linguistic expressions can be produced, the elaborated pragmatico-semantic analysis of which leads to proliferation problems.

It is also difficult to register the copies of multiplied entities in almost identical alternative models. We should apply safe and effective but very rapid methods in copying huge databases in a way that makes it possible for us to carry out the relevant differences between them.

Nevertheless, the most difficult task is the safe and systematic treatment of temporal entities, which come from the model of the external world as well as from the alternative models, and also come from the event structure of lexical items and from the discourse structure of sentences to be parsed. We have been led to the conclusion that the utter key to different kinds of systematization problems is utilizing points of time as special "stamps".

Let us then turn to the structure of the paper. Section II sketches the current version of ℜeALIS. Then our software ℜeALIS1.1 is demonstrated through discussing its different kinds of potential users (Section III) and its main use cases for the users we call internal users (Section IV) and for those we call external users (Section V). Section VI demonstrates the analysis of some linguistic examples with the purpose of elucidating our ambition to capture the highest possible level of human intelligence coded in language. We will conclude the paper (Section VII) with some remarks on the status of this "work in progress" among our previous works.

II. THE CURRENT VERSION OF ℜEALIS

ℜeALIS is a theory that immediately relies on Discourse Representation Theory (DRT) [1]. It can thus be introduced as belonging to the family of representational dynamic discourse semantics [1] [2]. Its (forty-page-long) definition is available at [23]. It is intended to reconcile the formal exactness of generative syntaxes with the dynamic approach of optimality theories and DRT, bearing in mind the holistic stance of cognitive linguists [3] [4].

In the post-Montagovian world of formal semantics, Discourse Representation Theory (DRT) [1]—which has offered a revolutionary solution to the resolution problem of ("donkey") anaphora and attractive visual representations for discourse meaning—is often criticized from "inside" as well as from "outside", considerably weakening its legitimacy. The internal criticism comes from the world of the dynamic model-theoretic semantics, from the Amsterdam School [5], and pertains to the (mathematically unquestionable) eliminability of exactly this attractive visual representation, insisting on "Montague's heritage" [6]. The external criticism comes from the Proof-Theoretic School [7], among others (Pollard [8]); they point at the dubious status and construction of possible worlds.

 ISBN: 978-1-61208-352-0

We claim that ℜeALIS [3] [4] [9]—while considerably relying on the representationalism of DRT in the course of solving a wide range of linguistic problems in order to maximally exploit and develop the excellent facilities provided by this representationalism—offers exactly the radical ontological innovation which lies with the elimination of the above-mentioned two dubious levels of representation, referred to as I and III below in Fig. 1.

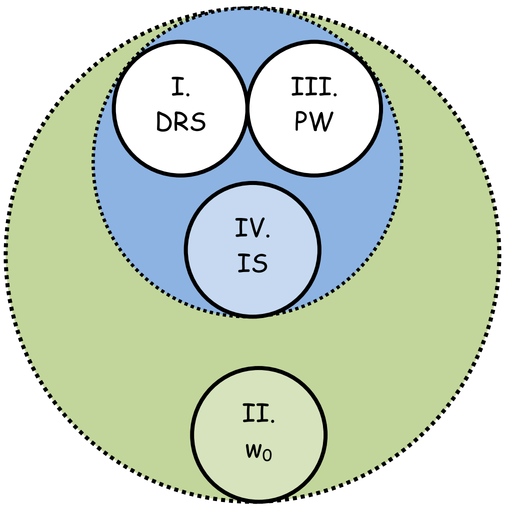


Figure 1. Components / levels of representation in DRT: I-IV; and their re-arranged ontology in ℜeALIS:
I. DRS: the semantic representation of sentences constituting coherent texts
II. Model of the external world (for extensional interpretation)
III. Possible worlds (for intensional interpretation)
IV. Interlocutors' information states

ℜeALIS embeds representational levels I and III—more exactly, their relevant content—in the representation of information states (IV), relying on the stance that, as interlocutors obtain information through discourses, their information states are worth regarding as gigantic, lifelong, DRSs. An information state has a double nature: it functions as a "representation" in the above regard while it is used as "what is to be represented" in the interpretation of, say, the intensional sentence types shown in (2b-d) below: it also depends on different persons' information states if these sentences are true, in contrast to sentence (2a), the truth value of which only depends on facts of the external world. Note in passing about the aforementioned "double nature of information states" that modern set theory exactly relies on a similar idea: Sets and their elements must not be mixed up; this does not mean, however, that a set could not serve as an element of another set.

a. "Ben is bald."
b. "Sue knows that/if [Ben is bald]"
c. "Joe guesses that Sue definitely wants to convince him to take it for granted that [Ben is bald]."

Figure 2. Sentences to be interpreted in different world(let)s.

We are going to illustrate the descriptive and explanatory power of ℜeALIS by sketching the interpretation of sentence (3a) below, featuring *realize*, which is a factive verb. Hence, it is a precondition of interpreting the sentence as true (or rather, as "well-formed") that the Evening Star should coincide with the Morning Star in (the model of) the external world. This means that the entity referred to as the Evening Star by the given astronomer should be the same entity he refers to as the Morning Star. In the approach of ℜeALIS, this relation is captured formally as demonstrated in (3b) below: the internal entity $r_{EveningStar}$ should be anchored to the same external entity as the internal entity $r_{MorningStar}$.

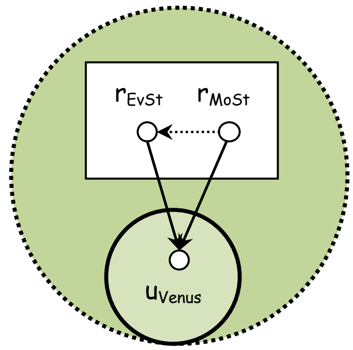


Figure 3. The interpretation of *realize* and the *Venus*-problem
a. "An ancient astronomer realized that the Evening Star is the same as the Morning Star."
b. $\alpha(r_{EvSt})$ is-the-same-as $\alpha(r_{MoSt})$ (since u_{Venus} is-the-same-as u_{Venus})
c. It does not hold that $r_{EveningStar}$ is-the-same-as $r_{MorningStar}$ at τ in the astronomer's worldlet of astronomic hypotheses
d. It holds that $r_{EveningStar}$ is-the-same-as $r_{MorningStar}$ at τ', which is a later point of time in the astronomer's worldlet of astronomic hypotheses

The astronomer himself is not (necessarily) aware of the co-anchoring of the two internal entities at his disposal (in his appropriate worldlet); but the fact of co-anchoring is an external requirement due to the factive character of the verb. Two further requirements to be satisfied in order for sentence (3a) to qualify as true concern two information states of the astronomer at different points of time, independently of the external world: what is to be checked is whether there is a "same-as" relation between the internal entity $r_{EveningStar}$ and the internal entity $r_{MorningStar}$ in the one information state (3d) while they do not stand in the "same-as" relation in the other one (3c)

All in all, three competing world(let) models should be considered simultaneously ("prism effect"), and three entities—an external one and two internal ones—should be inspected. As the three models are all parts of the one complete model of the history of the external world and all internal reflections associated with it (see Fig. 2 above), in this matrix model (3b-d) can all be checked.

It must be noted that the analysis relies on the same facilities available in the cognitive linguistics framework; see, for instance, Pelyvás [10], who follows Langacker's approach to nominal grounding [11]. The most important tenet of this view is that all nominals are grounded in the "reality" of the Idealized Conceptual Model(s) evoked in the discourse, which is relative to speaker and hearer, rather than directly in objective reality. From the point of view of linguistic analysis the reality that we could call "objective" (i.e., independent of speakers' and hearers' beliefs) is only of marginal importance...

III. USERS AND USES

A. Internal Users

Our software ℜeALIS1.1 is primarily intended to supply linguists with a device to build fragments of arbitrary languages of arbitrary morphological types. These fragments can capture such specialties of human languages as, for instance, the compositional cumulation of meaning units [6]. The definable meanings are pragmatico-semantic descriptions that satisfy the relevant definitions of ℜeALIS [3]. The group of users defined in Section I-A will be referred to as internal users.

B. External Users

Those using the developed language fragment will be referred to as external users. In the course of using the software, they can select lexical items to build sentences, the (generalized) truth-conditional interpretation of which they will be given on the basis of a "multiplied world model", which they have themselves constructed or received from "internal" experts [12].

Possible external users may be detectives or judges, for instance, who can have the truth of groups of propositions evaluated. In harmony with our "constructionist" stance, we mean by the aforementioned "generalized truth-conditional evaluation", besides the final *true/false* value, the collection of all the information required to reach this truth value. Our software thus, among others, serves the purpose of collecting and systematizing data in the effective structure ℜeALIS offers.

IV. USE CASES PROVIDED FOR INTERNAL USERS

A. Defining Relations

Internal users can define an external world w_0, over the universe of which (consisting of entities u_i) they can define relations of different arities [6]. One argument of all these relations is to be a series of disjoint temporal intervals. The software is to "dictate" (through permanent queries) the development of the external world: it requests new and new relations, and in the case of a given relation it requests the provision of (the initial and final points of) temporal intervals (among others).

Such relations can be defined in this way which are homogeneous in the sense that they qualify as true or false "momentarily", i.e., at each internal point of the temporal intervals independently. In Hungarian, for instance, *utazik* 'travel' and *úszik* 'swim' are homogeneous relations while *hazautazik* 'travel home' and *átússza* 'swim across' are heterogeneous. Further, each argument position of a relation can be associated with other relations of the group of relations defined earlier which provide us with restricting information. The agent argument of the Hungarian verb *utazik* 'travel', for instance, can be associated with the restricting relation *ember* 'human'.

B. Defining Label Strings of Worldlets

Relative to the set of "worldlets" (small partial models of alternative worlds) defined up to a certain point, the internal user can define (by simultaneous recursion) a new worldlet where the basis of this definition is the singleton consisting of the external world w_0. Specifically, relative to a worldlet w', a worldlet w'' can be determined through a quintuple of labels like the one shown in (4a) below. It defines the worldlet containing a human being's (r_{Sue}) knowledge ("maximal" belief); see sentence (2b) in Section II.

a. $\langle BEL,MAX,r_{Sue},\tau'',+\rangle$
b. $+/–/\varnothing/0/\theta$
c. $\langle\langle BEL,med,r_{Joe},\tau,+\rangle\langle INT,MAX,r_{Sue},\tau',+\rangle\langle BEL,MAX,r_{Joe},\tau'',+\rangle\rangle$

Figure 4. Labeling worldlets.

Alternatives to label BEL are labels INT (intention) and DES (desire), among others. Alternatives to label MAX are lower levels of intensity: e.g., aMX (almost maximal). The fourth member of the label quintuple is polarity; the values of this parameter are listed in (4b) above, but their interpretation is provided later.

The software can show through what kind of defining steps one can reach a worldlet relative to the external world as a fixed starting point. The label string in (4c) above, for instance, defines a worldlet which is to be regarded as the collection of information the status of which can be captured by means of the linguistic expression shown in (2c) in Section II.

C. Worldlets, Infons and Polarity Values

Internal users can assign pieces of information to worldlets. This procedure is to be "dictated" by the software as follows.

In the more general case, a point of time should be specified. As a reaction of the software, on the basis of the above-discussed temporal-interval series belonging to the relations, it is written which relations stand between which entities at the given point of time. If the user specifies, besides a point of time, a relation and some entities which occupy certain argument positions of the relation, the task of the software remains the writing of the lacking entities which stand in the given relation with the provided entities at the provided point of time. The unit of this writing process is the external infon [13]: an infon means the piece of information that certain entities stand in a certain relation at the given moment (e.g., Joe loves Sue, or Joe is just traveling).

Internal users can assign an infon (produced in the way sketched above) to an arbitrary worldlet for an arbitrary interval of time. The application of this temporal interval serves the purpose of capturing such factors as the dwindling into oblivion or some re-categorization of pieces of information.

Assigning a group E of infons to a worldlet standing with the external world in the relation provided in (4a) above can be interpreted as follows: Sue perceives information E from the external world and accepts as the current state of her environment. A similar interpretation in the case of the complex relation provided in (1c) is as follows: Joe suspects that Sue wants to make him to be sure that information E is true (while Sue herself, for instance, does not necessarily believe in the truth of E; nor is E true in the external world).

If the same infon is simultaneously assigned to someone's positive belief-worldlet (see '+' in (4b) above), negative desire-worldlet ('–' in (4b)) and neutral ('0' in (4b)) intention-worldlet, this complex "evaluation" captures this typical situation: the person in question perceives something and accepts its truth, but longs for its opposite without intending to change it (at least at that moment).

It is worth noting in connection with the polarity values listed in (4b) above that if an entity does not stand in the relation 'bald' in the external world, then the infon declaring the given entity's momentary baldness is to be assigned to the experiencer's negative ('–') or 'undefined' ('∅') belief-worldlets depending on the restricting relations, mentioned in Section II-A. Ben, for instance, can be thought by an experiencer to be "not bald" while in the case of the Eiffel Tower its baldness is undefined.

As for the crossed zero in (4b) above, the sentence variants shown in (2b) above illustrate its interpretation and significance. The variant with *that* can be captured by assigning the infon e declaring Ben's baldness to the worldlet (in the speaker's information state, i.e., mind) which contains the information that the speaker assumes Sue to qualify as true. The variant with *if*, however, should be accounted for in a slightly different way: what this variant reveals about the speaker' mind is that the speaker thinks that in Sue's mind infon e does not belong to a neutral belief-worldlet (one with a '0' polarity value).

D. Information Not Coming from Outside

Internal users can also assign information to worldlets indirectly, that is, not on the basis of (the relations of) the external world. This should be "dictated" by the software as follows.

The software should ask for predicate names and argument numbers, and then produce argument places with inserted "new" entities, which the software should also urge the user to anchor to "old" (external or internal) entities (NB their anchoring to any entities is only a possibility). Section II-C, where we defined the procedure of creating infons assigned to worldlets in human minds on the basis of states of affairs in the external world, is worth completing with a short comment. An internal infon does not contain the same entity names as the corresponding external infon does. Instead of identifying them, the correspondence between external and internal entities should be accounted for by anchoring elements of the former group to those of the latter group. In this way, we can explain the cases of misunderstanding where the same external fact is linked to different participants in two experiencers' minds.

E. Building the Lexicon

The internal user is given a core lexicon on the basis of the predicates the creation of which was described in II-C; and this core lexicon is enriched with the predicates created in the way described in II-D. Elements of the latter group of predicates should be associated with meaning postulates [6], by the help of queries of the software.

Note that items of the core lexicon need not be associated with meaning postulates since their interpretation is trivial on the basis of their creation: as they have been created by copying certain "patterns" of the external world, the rule concerning the pattern matching their semantic evaluation is based on is automatic. True perception and pattern matching is the same process, considered from opposite directions.

Let us return to the predicates whose forms are defined in II-D; they should be assigned meaning in the way to be defined in II-F. Before entering into details, it must be noted that this is the crucial innovation of $\Re$eALIS1.1, because this is the toolbox which exploits the advantages and results of all the model-theoretic theories, the discourse-representational innovations and the proof-theoretic ideas, and the "diagnosis" of cognitive linguistics on the weaknesses and shortcomings of these three approaches.

What comes from formal semantics [6]? The procedure of pattern matching. Further, the application of interpretational bases used as alternatives to each other ("possible worlds" $\rightarrow$ $\Re$eALIS-worldlets). And the consideration of the rate of successful instances of pattern matching compared to the entire set of possible instances of pattern matching.

The idea of operation over the partially ordered system of worldlets is due to Discourse Representation Theory [1] [2]. The step-by-step execution of this operation, referred to as 'accommodation' in DRT, coincides with the proof-theoretic processing of semantic information [7].

The modeling of the following linguistic elements is due to cognitive linguists [10] [11] [15]: *me, you, (s)he, here, there, now, then, these here* (in the context), *those there* (demonstration).

F. How to Define Lexical Items

The software should help the internal user (the formal linguist) in assigning (groups of alternative) phonetic forms and meaning postulates to predicate names, besides such straightforward information as (sub)categorization and argument number.

Meaning postulates essentially consist of first-order formulas. The most peculiar element of our method is that each formula like this should be associated with a set of such chains of worldlet labels as the one shown in (1b) above and the information as to which worldlet(s) these chains to be linked to in the course of interpreting sentences (possibilities are the external worldlet, certain worldlets of the selected speaker, addressee, or participants referred to in the sentences, or worldlets which can be identified in the selected context or scope of demonstration (see the last paragraph in Section II-E).

V. USE CASES FOR EXTERNAL USERS

The external users—who can construct a sentence and specify the speaker, the addressee, the entities assumed to be present in the context (and possibly a subset of those in the scope of some demonstration), the speech time and the time of reference, among others—are given a generalized truth evaluation. This means that they are given not only a truth value but also all the pragmatic well-formedness conditions of the sentence "performed" in the specified context.

Thus, they can look "inside" all relevant participants' minds (i.e., the current and possibly some previous information states). They can realize, for instance, if the definite noun phrases are suitable for unambiguously identifying the intended *denotata*. They can also receive information about the success of satisfying other kinds of presupposition. They can detect, through comparing the information provided by the sentence and the information found in the specified interlocutors' appropriate worldlets, if there might emerge some misunderstanding, lie, bluff, deception [4].

VI. Linguistic Examples

A. *Generalized Truth Evaluation*

External users are given a peculiarly multiplied data base which contains, besides a relational model of some fragment of (the history of) the external world, several of its alternatives. These alternatives essentially play the role of possible worlds, known from intensional model-theoretic semantics, but they are finite constructions appearing as such parts of information-state models of interlocutors that can be construed as their beliefs, desires, intentions, or any other kinds of fictions (II-B [1]).

This arrangement of worldlets enables us to carry out truth evaluation not only on the basis of the external world, which is necessary and sufficient, for instance, in the case of sentences (5a-a') below, but also on the basis of internal worldlets, which is obviously necessary in the case of sentences like (5b). The truth of the variants shown in (5b) does not depend on any facts in the external world. It depends on nothing else but Joe's knowledge, or the knowledge that Sue attributes to Joe. In this latter case, it requires more steps to reach the worldlet which can serve as the basis of truth evaluation (external world model → Sue's belief → Sue's hypotheses on Joe's beliefs); cases like this make it necessary to localize worldlets in the recursive way illustrated in (4c) in II-B. Verbs expressing modal attitude (e.g., *think*, *guess*, *conjecture*, *wish*) and many other expressions (e.g., *according to someone*) can be associated with meaning postulates by means of the tool described in II-F: the essence of their meanings lies with the "direction indicator" function. Such direction indicators help us with finding the worldlets which can serve as the basis of the truth evaluation of the proposition that appears in the appropriate argument positions of the modal verbs or other linguistic expressions in question.

a. It was snowing. a'. It has snowed.
b. (Sue thinks that) Joe knows that it was snowing.
c. Patty was traveling home.
d. *That tall Finnish woman* is pretty.

Figure 5. Generalized truth evaluation relying on worldlets.

B. *Past Continuous and Present Perfect*

Internal users can work out exacting and sophisticated syntaxes and semantics by the help of the toolbox offered by $\mathfrak{R}$eALIS1.1.

The truth value of (5a) above, for instance, can be calculated in the following way: the software should query the values of *then* and *there*, and then it should localize the area of the temporal external world model where pattern matching should be attempted in order to decide if it is snowing "then" and "there" (III).

The truth evaluation of (5a'), however, requires the values of *here* and *now*, and what is to be checked in the external world is whether the landscape is snowy. The meaning postulate of the verb *snow*, thus, contains the determination of the result state (*snowy*), too. Note in passing that (5a') pragmatically suggests that it is not snowing at the relevant moment while the land is snowy as a result of an earlier snowing.

C. *Progressive Aspect*

The truth evaluation of sentence (5c) above also requires a polished and exacting meaning postulation because not only facts of the external world should be taken into account. A progressive sentence like this is also to be evaluated to be true in a case in which Patty never got home but she proves to have been travelling at the moment of *then*, she proves to intend to come home, and the speaker proves to attribute a quite high likelihood to this arrival (II-F) [1]. Thus, the content of certain internal worldlets is to be checked, besides the partial satisfaction of a travelling event in the external-world model.

D. *The Intensional Character of Nicknames*

A demanded pragmatico-semantic analysis of nicknames also requires the toolbox sketched in II-F. Who is *Patty* in (5c) above, for instance? Internal users can capture the essence of the task of finding *denotata* by construing nicknames as special predicates the "truth evaluation" of which involves not (only) the external control on the correspondence between official names and nicknames but (also) the worldlets concerned in the following questions: is *Patty* a possible nickname of the speaker for the given person, does the speaker think that the addressee may (also) call her Patty, do they know this about each other, and so on. Hence, internal worldlets are to be checked via pattern matching.

E. *(Partially) Subjective Predicates*

Example (5d) above illustrates further advantages in meaning postulation of the toolbox demonstrated in II-F. The adjective *pretty*, for instance, is worth regarding as a fully personal and subjective judgment, with no extension in the external world. Nevertheless, (5d) does mean exactly the same as the sentence *I consider her pretty*. The truth of this latter sentence exclusively depends on the speaker while it would be elegant to base the evaluation of sentence (5d) on a somewhat less speaker-dependent calculation. As follows, for instance: (5d) is considered true if *most* persons in the external-world model consider the given lady to be pretty. According to an even more elegant solution, instead of the entire set of persons, only those *respected* by the speaker are considered. The extension of the verb respect is to be checked in the external-world model.

F. Demonstration and Anchoring

The demonstrative noun phrase in (5d) illustrates another instance of the necessity for "pragmatically conscious" truth evaluation. *That* asks for the value of the "those there" parameter from the external user. It is elegant to assume that this value is a set of entities, out of which the software should select a unique entity on the basis of the predicates *tall*, *Finnish* and *woman*. Their extensions count in the external world, at least primarily; it is an elegant facility, however, to inspect the speaker's beliefs as well, or the speaker's hypothesis about the addressee's beliefs: sentence (5d) can be evaluated as *true but ill-formed* if, for instance, the speaker intends to refer to a tall Swedish woman about whom they think, incorrectly, that she is Finnish.

VII. CONCLUDING REMARKS

The current implementation of ℜeALIS1.1 is a client-server Windows application that has been elaborated in a Delphi environment, which guarantees rapid and flexible development. Access to data is executed via standard SQL commands by means of a relational data-base management system. For this purpose, we currently use Firebird Interbase.

The Prolog basis, applied in the experimental phase of our research [16] [14] [17] [18] [19] [20] [21] [22], has been replaced with Delphi environment, which is more capable of managing large data-bases, developing user-friendly interfaces, and constructing more complex applications. This the radical difference between ℜeALIS1.1 and the aforementioned works.

The menu items correspond to the services sketched in Section IV. Particular menu items are available to the different kinds of users defined in Section III after checking their identity and authenticity. At some points, the program provides illustrations of the structures constructed by either the system or its users: for instance, parsing trees, systems of worldlets, anchoring relations of entities. Figure 6 below illustrates this last facility.

The software is permanently developed and expanded, exploiting new scientific results; it has repercussions on the theory, due to the fact that the theory can be tested by means of the software.

As our software inherently belongs to a radically new and holistic "pragmalinguistics" theory (Section II), it would be uneasy to compare to softwares based on some different theoretical foundation. We are working on developing tests to evaluate its effectivity.

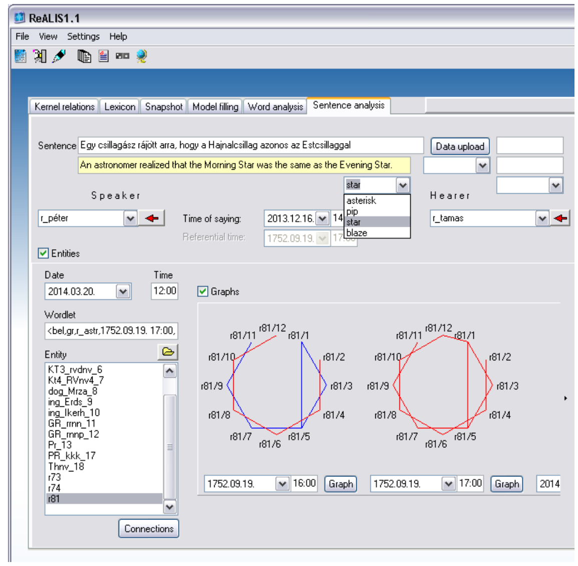


Figure 6. Interpreting the sentence shown in Fig. 3.

ACKNOWLEDGMENT

We are grateful to SROP-4.2.2.C-11/1/KONV-2012-0005 (Well-Being in the Information Society).

REFERENCES

[1] H. Kamp, J. van Genabith, and U. Reyle, "Discourse Representation Theory," in Handbook of Philosophical Logic, vol. 15, D. Gabbay and F. Guenthner, eds. Berlin: Springer, 2011, pp. 125–394.

[2] N. Asher and A. Lascarides, Logics of Conversation. Cambridge: Cambridge Univ. Press, 2003.

[3] G. Alberti and M. Károly, "Multiple Level of Referents in Information State," LNCS7181, 2002, pp. 349–362.

[4] G. Alberti, N. Vadász, and J. Kleiber, "Ideal and Deviant Interlocutors in a Formal Interpretation System," to appear in The communication of certainty and uncertainty, A. Zuczkowski, ed. Amsterdam: Benjamins, 2014.

[5] J. Groenendijk, M. Stokhof, and F. Veltman, "Coreference and Modality," in The Handbook of Contemporary Semantic Theory, Sh. Lappin, ed. Oxford: Blackwell, 1996, pp. 179–213.

[6] D. R. Dowty, R. E. Wall, and S. Peters, Introduction to Montague Semantics. Dordrecht: Reidel, 1981.

[7] F. Nissim and R. Dyckhoff, "Proof-theoretic semantics for a natural language fragment," Linguistics and Philosophy, vol. 33 (6), 2010, pp. 447–477.

[8] C. Pollard, "Hyperintensions." ESSLLI 2007, http://www.cs.tcd.ie/esslli2007, 2007.

[9] J. Kleiber and G. Alberti, "Uncertainty in Polar Questions and Certainty in Answers?," to appear in Certainty-uncertainty – and the attitudinal space in between, S. Cantarini, W. Abraham, and E. Leiss, eds. Amsterdam: Benjamins, 2014.

[10] P. Pelyvás, "Relating Cognitive Models In Nominal Grounding. Referential, Attributive Use and Referential Opacity: A Case for a Blend?" in Metaphors of sixty. Papers presented on the occasion of the 60th birthday of Zoltán Kövecses, R. Benczés and Sz. Csábi, eds. Budapest: SEAS, ELTE, 2006, pp. 196–209.

[11] R. W. Langacker, "Remarks on Nominal Grounding," *Functions of Language* 11:1. 2004, pp. 77–113.

 ISBN: 978-1-61208-352-0

[12] I. Kilián and G. Alberti, "A Metamodel-Driven Architecture for Generating, Populating and Manipulating "Possible Worlds" to Answer Questions," in ICSOFT 2013 Proceedings, July 2013, pp. 74-78.

[13] J. Seligman, J. and L. S. Moss, "Situation Theory," in Handbook of Logic and Language, J. van Benthem and A. ter Meulen, eds. Amsterdam: Elsevier, and Cambridge: MIT Press, pp. 239–309, 1997.

[14] G. Alberti, J. Kleiber, and A. Viszket, "GeLexi Project: Sentence Parsing Based on a GEnerative LEXIcon," Acta Cybernetica 16 (Hungary). 2004, pp. 587–600.

[15] R. W. Langacker, "Cognitive Grammar," in The Oxford Handbook of Cognitive Linguistics. Oxford: Oxford Univ. Press, 2007, pp. 421–462.

[16] G. Alberti, K. Balogh, J. Kleiber, and A. Viszket, "Total Lexicalism and GASGrammars: A Direct Way to Semantics," LNCS 2588, 2003, pp. 37–48.

[17] G. Alberti, and J. Kleiber, "The *GeLexi* MT Project," in Proceedings of EAMT 2004 Workshop (Malta), J. Hutchins, ed. Valletta: Univ. of Malta, 2004, pp. 1–10.

[18] G. Alberti and J. Kleiber, "The Grammar of $\mathfrak{R}$eALIS and the Implementation of its Dynamic Interpretation," Informatica (Slovenia) 34/2, 2010, pp. 103–110

[19] G. Alberti, M. Károly, and J. Kleiber, "From Sentences to Scope Relations and Backward," in Natural Language Processing and Cognitive Science, Proceedings of NLPCS 2010, B. Sharp and M. Zock, eds. Funchal: SciTePress, 2010, pp. 100-111.

[20] G. Alberti, M. Károly, and J. Kleiber, "The $\mathfrak{R}$eALIS Model of Human Interpreters and Its Application in Computational Linguistics," in Software and Data Technologies: 5th International Conference, ICSOFT, J. Cordeiro, M. Virvou, and B. Shishkov, eds. Heidelberg: Springer, 2010, 468–474.

[21] G. Alberti and M. Károly, "The Implemented Human Interpreter as a Database," in Proc. of Int. Conf. on Knowledge Engineering and Ontology Development, J. L. G. Diets, ed., Funchal: SciTePress, 2011, pp. 379–385.

[22] M. Károly and G. Alberti, "The Implementation of a $\mathfrak{R}$eALIS-based Method of Static Intensional Interpretation," in Proc. of 5th International Conference on Knowledge Engineering and Ontology Development, J. Filipe and J. G. L. Dietz, eds, 2013, pp. 393–398.

[23] http://lingua.btk.pte.hu/realispapers

 ISBN: 978-1-61208-352-0

A Comparative Study of Imputation Methods in Predicting Missing Attribute Values in DGA Datasets

Zahriah Sahri
Fakulti Teknologi Maklumat dan Komunikasi,
Universiti Teknikal Malaysia Melaka,
Melaka, Malaysia.
szahriah2511@yahoo.com.my

Rubiyah Yusof
Malaysia Japan Internatioanl Institute of Technology,
Universiti Teknologi Malaysia,
Kuala Lumpur, Malaysia.
rubiyah@ic.utm.my

***Abstract*—Dissolved Gas Analysis (DGA) is one of the most deployable methods for detecting and predicting incipient faults in power transformers. For predicting faults, DGA uses tools such as Doernenburg, Rogers and IEC methods. The presence of missing values in a DGA dataset may affect the diagnostic performances of these three methods. This study applies the mean, regression, expectation-maximization, multiple imputation, and *k*-nearest neighbor methods to replace the missing values with estimated values in a DGA dataset. Using the number of unresolved diagnoses, the number of wrong diagnoses, and the number of correct diagnoses as the criteria to evaluate the effects of the imputation methods on the DGA diagnostic methods, this study shows that *k*-nearest neighbor increases the performances of Doernenburg, Rogers and IEC methods the most on two datasets with actual missing values. Experimental results show that imputing missing values in DGA datasets has increased diagnostic performance of the three ratios methods of DGA**

Keywords-dissolved gas analysis; missing values; imputation methods; gas ratios method; fault diagnosis

I. INTRODUCTION

Power transformers are a must-have item for any power utility company to increase or decrease electrical power for transmission and distribution throughout interconnected power systems. Because of prolonged usage and as transformers age, their internal conditions degrade when faults such as short-circuit, arcing, partial discharges or overheating occur during operations. These faults will release several gases commonly known as the fault gases: hydrogen (H_2), acetylene (C_2H_2), ethylene (C_2H_4), methane (CH_4), ethane (C_2H_6), carbon monoxide (CO) and carbon dioxide (CO_2) that stay dissolved at above threshold values in the insulating oil of a transformer. A faulty transformer must be removed from operation, sent for repair and/or replaced. These processes are costly and time-consuming but are necessary because if left unattended for long, a faulty transformer may trigger worse impacts such as explosions, loss of human lives, or environmental disasters. Recognizing the need for checking the serviceability of transformers, most utility companies perform preventive maintenance either periodically or conditionally to detect these faults.

Among the many existing techniques to detect these faults, Dissolved Gas Analysis (DGA) is recognized as the most effective method and is practiced universally today [1]. It involves a few sequence processes as follows: a) sampling of oil from the transformer, b) extracting the fault gases dissolved in the oil, c) calculating and analyzing the concentration of these gases, their gassing rates and the ratios of certain gases d) finally, the identification of the fault types. Currently, traditional diagnostic tools such as Key Gas [1], IEC ratios [2], Rogers ratios [1], Doernenburg ratios [1] and Duval Triangle [3] are widely used in the fourth process of DGA method. These ratio methods identify fault types using the ratios of certain fault gases and each ratio is assigned to one or more numerical thresholds. These thresholds are coded and mapped to specific faults. However, in some cases, gas concentrations may be *incomplete* which lead to combination of ratios that do not match any predefined threshold. As a result, fault that occurs inside a transformer may be classified unknown or inconclusive - a well-known shortcoming of the DGA ratios methods as documented in [4].

One of the reasons for incomplete gas concentrations is *missing values* for some of the fault gases. Missing values in DGA occur for various reasons, such as acetylene evaporates quickly, or the existence of contamination on the surface of the platinum alloy of a gas meter, and some transformer faults generate only a few fault gases [2]. Such samples containing missing values were usually manually deleted and excluded from subsequent analysis [5-6]. Majority reported only complete-case samples of DGA data [7-9]. Only researchers in [10] estimated the missing values in a DGA dataset using Support Vector Machine regression, which increased the accuracy of their Naive-Bayes classification algorithm. However, there are very few published literature concerning missing value estimation for DGA data especially with the objective of improving the diagnostic performance of the gas ratio methods. On the other hand, certain fields such as microarray analysis [11-12], medical [13-14], and social sciences [15-16] have paid more attention to this issue. Consequently, statistical analyses or machine learning algorithms that were subsequently applied after filling in the missing values have shown better results.

The aforementioned scenario motivates us to fill-in the missing values in DGA datasets with estimated values ("imputing") to minimize the inconclusive diagnoses of the gas ratio methods. Fortunately, there are many imputation methods to choose from [11,17-19], ranging from simple to complex solutions, and statistical to machine learning methods. In this paper, we propose imputing missing values in a DGA dataset using a few established methods Mean/mode (MEAN), linear regression (REG), expectation-maximization (EM), multiple imputation (MI) and *k* nearest neighbour algorithm (*k*NN) are the selected imputation methods, and Rogers ratios, Doernenburg ratios, IEC ratios are the DGA ratios methods to diagnose the transformer faults. A "before and after" experiment is conducted to compare the diagnostic performance of each ratio method applied on the original datasets with missing values and that applied on the complete datasets imputed by each imputation method.

In Section II, we provide related work on missing values and the application of imputation methods. Section III we briefly described the DGA method and its ratio-based diagnostic methods. The compared imputation methods are described in Section IV. Section V presents and analysed experimental results. Section VI concludes our findings.

II. Related Work

According to [17], a missing value indicates a lack of response. “Don’t know”, “Refused”, “Unintelligible”, and “Nil” are some of the possible codes for missing values in a dataset. However, why worry about missing values? Two major negative effects are reported in [20]. First, missing values reduce statistical power. Second, missing values could result in biased statistical estimates in several ways. Before deciding which imputation are suitable for the incomplete datasets, researchers should ask whether the pattern of missing observations is random or not ("patterns of missingness"). Little and Rubin [21] distinguished randomness into the following categories: Missing Completely at Random (MCAR), Missing at Random (MAR), and Not Missing at Random (NMAR).

MCAR exists when value(s) are missing because of uncontrolled events, which are totally independent of the potential values of both the observed and the missing variables. In the case of DGA data, acetylene is a very soluble and reactive gas, and disappears fast from absorption; thus, acetylene sometimes does not appear in the collection sample. On the other hand, when data are MAR, the probability of a missing value on some variables is dependent on the value of the observed variables. However, the missing value itself is not the cause for the missing. For DGA data, a fault such as high-energy arcing releases high quantities of hydrogen and acetylene. Such a pattern is considered to be MAR, and most missing data treatment methods are assumed to be MAR Finally, NMAR occurs when the probability of missing a value is a function of the value itself. NMAR is very unlikely to appear in DGA data; because gases are released as a result of faults in power transformer.

Statistical analysis with missing data has been noted in the literature for more than 70 years. Walks [22] initiated a study on the maximum likelihood estimation for multivariate normal models with fragmentary data. Thereafter, extensive discussions on this topic continue. A useful reference for general parametric statistical inferences with missing data can be found in Little and Rubin [23]. Litwise deletion (LD), pairwise deletion (PD), REG, MEAN, EM, and MI are some statistical tools available for imputing missing values. LD and PD are deficient in several aspects. Despite their simplicity, both are inefficient [24-25]. LD, in particular, can discard an enormous amount of potentially useful data. PD may be more efficient than LD in many cases, but for some data structures it is known to be less efficient [26]. MEAN, because of its simplicity, is commonly used in the social sciences as a fast alternative to LD [14]. Also, it is often used as a base for other proposed imputation methods such as in [27-28]. EM and MI, currently represent the state of the art, have been applied to various problem domains. Researchers in [29] estimated the missing values of leaf area index , a biophysical variable, using EM and helped reduce the root mean square error of the Gaussian Bayes Network output. In the medical field [13], MI was found to preserve observed and real data better than complete-case and dropping a particular variable approaches when predicting for deep venous thrombosis in patients.

Machine learning algorithms have also garnered large followings as the choice for data imputation. The Naive Bayes classifier, which is the least sensitive to missing data, learns effectively without the need to treat missing values, especially if the missing rate is less than 30%. It makes full use of the observed data and, thus, is the most adaptive to missing data [30]. In [31], *k*NN was used to impute missing values in software project datasets. They found that the k-NN imputation can improve the prediction accuracy of C4.5, particularly if the missing data percentage exceeds 40%. A comparison of various imputation machine learning methods has also been published [32]. They concluded that self-organizing map and multi-layer perceptron performed slightly better than regression-based imputation and nearest neighbour method. Comparisons were performed between statistical and machine learning imputation methods in [14] for imputing missing values in breast cancer datasets. Their findings show that machine learning based imputation methods outperformed the statistical ones.

III. Dissolved Gas Analysis

During normal operation, oil-insulated power transformers produce gases such as hydrogen and hydrocarbon, albeit in very small quantities. However, when they experience electrical disturbances or thermal decomposition, the chemical reactions of the insulation involves the breaking down of carbon-hydrogen and carbon-carbon bonds. During this process, active hydrogen atoms and hydrocarbon fragments are formed. These fragments can combine with each other to form gases: hydrogen, methane, acetylene, ethylene, ethane, carbon monoxide, and carbon dioxide. These gases are usually referred to as the fault gases, which stay dissolved at above threshold values in the presence of fault(s). The composition of these dissolved gases is dependent on the type of faults that occur as shown in Table 1 of some common power transformer faults.

DGA uses this strong relationship between certain combination of dissolved gases and their associated fault as the underlying principle in identifying incipient faults in power transformer. In general, a DGA dataset is built by taking oil samples over a period of time at regular interval to discern trends and to determine the severity and progression of incipient faults. Generally, daily or weekly sampling is recommended after start-up, followed by monthly or longer intervals. Routine sampling intervals may vary depending on application and individual system requirements [1].

This study briefly describes the ratio-based diagnostic methods mentioned in the Section I that map the dissolved gases found in the oil sample with corresponding fault. Compared to other tools, these ratios methods have become a standard to diagnose fault based on DGA results [1]. These methods employ an array of ratios of certain key combustible gases as the fault type indicators. These five ratios are:

Ratio 1 (R1) = CH_4/H_2
Ratio 2 (R2) = C_2H_2/C_2H_4
Ratio 3 (R3) = C_2H_2/CH_4
Ratio 4 (R4) = C_2H_6/C_2H_2
Ratio 5 (R5) = C_2H_4/C_2H_6

TABLE 1. COMMON TYPES OF TRANSFORMER FAULTS AND THE KNOWN FAULT GASES ASSOCIATED WITH THEM

No	Gases Present Prominently During Operation	Interpretations
1		Normal operation
2	Nitrogen, carbon monoxide, and carbon dioxide	Transformer winding insulation overheated: key gas is carbon monoxide
3	Nitrogen, ethylene, and methane, some hydrogen and ethane	Transformer oil is overheated: key gas is ethylene
4	Nitrogen, hydrogen small quantities of ethylene and ethane	Corona discharges in oil: key gas is hydrogen
5	Same as item No. 4 with the inclusion of carbon dioxide and carbon monoxide	Corona involving paper insulation: key gas is hydrogen
	Nitrogen, high hydrogen and acetylene, minor quantities of ethylene and methane	High-energy arcing: key gas is acetylene
	Same as item No. 6 with the inclusion of carbon dioxide and carbon monoxide	High-energy arcing involves paper insulation of winding: key gas is acetylene

A. Doernenburg Ratios

This method suggests the existence of three general fault types namely, thermal, partial discharge, and arcing. This method requires significant levels of the gases to be present in order for the diagnosis to be valid. The method utilizes Ratios 1, 2, 3, and 4 and the ratios in the order Ratio 1, Ratio 2, Ratio 3, and Ratio 4 are compared to limiting values, providing a suggested fault diagnosis as given in Table 2.

B. Rogers Ratios

This method utilizes Ratios 1, 2, and 5. The method does not depend on specific gas concentrations to exist in the transformer for the diagnosis to be valid. However, it suggests that the method be used only when the normal limits of the individual gases have been exceeded. Table 3 gives the values for the three key gas ratios corresponding to suggested diagnoses.

C. IEC Ratios

This method also utilizes the same three Ratios 1, 2, and 5 as in the revised version of Rogers Ratios. The key differences are the range of code assigned to each ratio and the number of suggested faults. The IEC Ratios fault interpretations can be divided into 9 different types and they are shown in Table 4.

IV. METHODOLOGY OF STUDY

This study imputes the missing values found a DGA datasets using the four established methods mentioned in the Introduction section with the objective of reducing the inconclusive diagnoses of the gas ratio methods. Fig. 1 shows the methodology of this study in meeting this objective.

A. Mean Imputation

One of the most frequently used methods. This method consists of replacing the missing data for a given attribute by the mean (quantitative attribute) or mode (qualitative attribute) of all known values of that attribute. This study chose the mean because DGA datasets contains continuous variables. It is easy to execute as well as suitable for all patterns of missingness. However, this approach also introduces biases. The main reasons are that it shifts the possible extreme values back to the middle of the distribution, and it reduces variances in the variable being imputed. Sample sizes may be overestimated and correlations may be negatively biased, also, when applying this technique [17].

B. Regression Imputation

This can be accomplished by regressing the variable with missing data on other variables in the data set for those cases with complete data. The estimated regression equation is then used to generate predicted values for the cases with missing data. This technique assumes that the data are MAR or MCAR, which makes it suitable for imputing DGA

TABLE 2. DORNERBURG RATIOS [1]

Suggested Fault Diagnosis	R1 $CH4/H2$		R2 $C2H2/C2H4$		R3 $C2H2/CH4$		R4 $C2H6/C2H2$	
	Oil	Gas Space	Oil	Gas Space	Oil	Gas Space	Oil	Gas Space
Thermal decomposition	>1.0	>0.1	<0.75	<1.0	<0.3	<0.1	>0.4	>0.2
Partial Discharge (low intensity PD)	<0.1	<0.01	Not significant		<0.3	<0.1	>0.4	>0.2
Arcing (high intensity PD)	>0.1 to <1.0	>0.01 to <0.1	>0.75	>1.0	>0.3	>0.1	<0.4	<0.2

TABLE 3. ROGERS RATIOS [1]

R1 CH_4/H_2	R2 C_2H_2/C_2H_4	R5 C_2H_4/C_2H_6	Suggested Fault Diagnosis
<0.1	>0.1 to <1.0	<1.0	Unit normal
<0.1	<0.1	<1.0	Low-energy density arcing-PD
0.1 to 3.0	0.1 to 1.0	>3.0	Arcing-High energy discharge
<0.1	>0.1 to <1.0	1.0 to 3.0	Low temperature thermal
<0.1	>1.0	1.0 to 3.0	Thermal fault < 700 ^oC
<0.1	>0.1	>3.0	Thermal fault >700 ^oC

datasets. There is, however, a general tendency to produce underestimates of variances and overestimates of correlations [33].

C. *Expectation Maximization*

This algorithm was introduced by [34]. It capitalises on the relationship between missing data and the unknown parameters of a data model. The basic algorithm consists of two steps: expectation (E step) and maximization (M step). First, separate the data into missing and observed, and establish starting values for the parameters (mean, variance, and covariance). In the E step, using these parameters, compute the predicted scores for the missing data (the expectation). In the M step, using the predicted scores for the missing data, maximize the likelihood function to obtain new parameter estimates. Repeat the process until convergence is obtained. EM assumes that the data are MAR. EM requires a fairly large sample for the estimates to be approximately unbiased and normally distributed [17].

D. *Multiple Imputation*

This method was proposed by Rubin [35] in 1987. The whole MI procedure is made of three steps. They are imputation, analysis, and pooling processes. We applied only the first step, imputation process, of the three steps because our interest is to fill in missing values with estimated values. This simulation method replaces each missing value with $m > 1$ plausible values, which are drawn randomly from their predictive distribution. m is the number of repetition. Imputing a missing value with m simulated values produces m apparently completed datasets and then the mean of m imputed values was filled in the missing value. Post-imputation, MI allows analysts to proceed with familiar complete-data techniques and software. Another positive point is that a large number of repetitions is not always necessary to obtain precise estimates. For example, with 50% missing information, $m = 10$ imputations is efficient. However, like EM, MI does rely on large sample approximations but works better than EM in small to moderate sample sizes. MI also assumes that the missing data are MAR [17].

E. *kNN Imputation*

We use the k-NN algorithm to determine the imputed data, where nearest is usually defined in terms of a distance function based on the auxiliary variable(s). In this method a pool of complete instances is found for each incomplete instance, and the imputed values for each missing cell in each recipient is calculated from the mean or median of the respective attribute in complete instances. Mean is used with continuous attributes, whilst median is suitable for discrete attributes. This study chose mean as DGA dataset contains continuous variables. For the kNN method, two parameters need to be determined for achieving high estimation accuracy: the number of nearest neighbour (k) and the distance metric: the choice of k, the number of neighbours used and the appropriate distance metric. Simulation results have demonstrated that for small datasets, $k = 10$ is the best choice [36], while [37] observed that k is insensitive to values of k in the range of 10-20. Therefore, this study replaced the missing values with estimated values from 1-10 nearest neighbours depending on the size of datasets. The distance metric used was the Euclidean distance as adopted by [37].

The advantages of the kNN imputation are [31]:

- It does not require to create a predictive model for each feature with missing data.
- It can treat both continuous and categorical values.
- It can easily deal with cases with multiple missing values.
- It takes into account the correlation structure of the data.

TABLE 4. IEC RATIOS

R1 CH_4/H_2	R2 C_2H_2/C_2H_4	R5 C_2H_4/C_2H_6	Suggested Fault Diagnosis
0.1 to 1.0	<0.1	1.0 to 3.0	Thermal fault < 150 ^oC
>1.0	<0.1	<1.0	Thermal fault 150^oC - 300 ^oC
>1.0	<0.1	1.0 to 3.0	Thermal fault 300^oC - 700 ^oC
>1.0	<0.1	>3.0	Thermal fault > 700 ^oC
0.1 to 1.0	1.0 to 3.0 and >3.0	1.0 to 3.0 and >3.0	Discharge of low energy
0.1 to 1.0	1.0 to 3.0	>3.0	Discharge of high energy
<0.1	<0.1	<1.0	Partial discharge of low energy density
<0.1	1.0 to 3.0	<1.0	Partial discharge of high energy density
0.1 to 1.0	<0.1	<1.0	normal

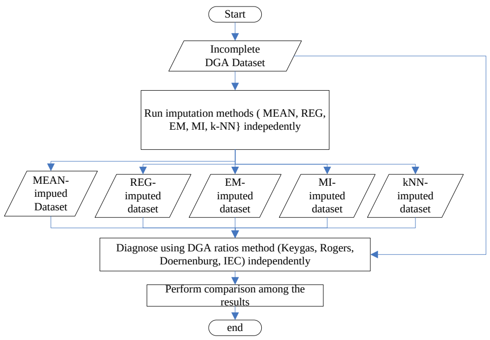


Figure1. Comparative evaluation of imputation methods

V. EXPERIMENTS AND RESULT

A. *DGA Datasets*

Two DGA datasets, with different percentages of actual missing values, are imputed and classified in this study. The first DGA dataset (named MAL) is obtained from a local Malaysian utility company which manage various transformers located throughout Malaysia, whilst IECDB10 [38] is the second. The characteristics of the datasets are shown in Table 5. A sample of DGA data consists of a number of dissolved gases in oil and the corresponding fault type as shown in Table 6. Dashes in Table 6 represents missing values (missing gases). Using a gas chromatograph equipped with suitable adsorption columns, the dissolved gas concentrations are measured from the oil samples in parts per million (ppm) by volume of (specific gas) in oil.

B. *Experimental setup*

This study used SPSS [39] to impute missing values using the MEAN, REG, and EM methods. Meanwhile the MI and *k*NN methods were applied using SOLAS and MATLAB [40], respectively. For the EM method, a normal distribution (the default) of the data was assumed and the default iterations (25) was adopted. This study chose normal variates as the random component to be added for the REG estimation task. For the MI method, this study chose $m = 5$ because according to [16], the MI method does not need a large number of repetitions for precise estimates. Because both datasets in Table 5 were quite small in size, this study chose *k*=1,3,5,7,10 as explained in Section IX.E.

TABLE 5 THE CHARACTERISTICS OF DGA DATASETS USED IN THIS STUDY

	IEC10DB	**MAL**
Number of samples	167	1228
Number of dissolved gases	7	9
Number of fault type	6	6
Instances with missing values (%)	27.54	76.07
Missing values (%)	7.96	14.21

TABLE 6. DGA DATASET WITH MISSING VALUES (ppm)

H2	**O**	**N2**	**CH4**	**CO**	**CO2**	**C2H4**	**C2H6**	**C2H**	**Fault**
9	3419	29804	5	403	1316	12	1	-	PD
6	5830	50411	-	217	2039	1	-	-	PD
52470	33164	94134	2504	12515	566	640	18	3251	Arcing
-	13877	54589	3	624	2043	56	3	-	Arcing

After independently filling in the missing values in DGA datasets using the compared methods, the three ratios methods were then run independently on each imputed dataset. The effectiveness of each imputation method were evaluated based on the number of unresolved diagnoses reported by each ratio method, the number of wrong diagnoses made by each method and the number of correct diagnoses predicted by each method.

C. *Case Study 1:IEC10 Dataset*

Table 7 shows the result of diagnoses for each *k* of the kNN method on the IEC10DB dataset for each DGA diagnostic method. It is observed that when *k*=1, all of the three ratios method gave the highest number of correct predictions. Similar observation was seen for the number of wrong diagnoses, all three methods were the least wrong when *k*=1. For the number of unresolved diagnoses, the Rogers and IEC reduced the unresolved cases the most when *k*=1, but Doernenburg did the same when *k*=5. Overall, higher values of *k* do not increase diagnostic performance of the ratio methods on this dataset. It can be explained that the inclusion of complete instances that are significantly different from the instance of interest can decrease accuracy because the neighborhood has become too large and not sufficiently relevant for the estimation task.

Table 8 shows the result of diagnoses for each ratio method on imputed datasets obtained from each imputation method compared in this study. INCOMPLETE is the results of diagnoses from each ratio method before the missing values were estimated by the comparative methods. For the *k*NN method, the result from *k*=1 was used for comparison with the other imputation methods. It can be safely said, for this dataset, replacing missing values with estimated values from each imputation method reduce the unresolved diagnoses compared to INCOMPLETE. The MEAN was the most effective in reducing the number of unresolved diagnosis for Doernenburg. For Rogers and IEC, it was the *k*NN. Consequently, the reduced statistics of unresolved lead to an increase or decrease of wrong and correct predictions for all of the ratio methods. It is seen that, among the compared imputation methods, *k*NN registered the highest number of correct guesses and the least number of wrong diagnoses. Interestingly, four methods (kNN, EM, MI, REG) helped increase the diagnostic performance of all the ratios method in guessing correct faults compared to INCOMPLETE. MEAN however fared worst than INCOMPLETE when applied with Rogers and IEC but did better than INCOMPLETE in Doernenburg. Unfortunately, the number of wrong diagnoses increased for all ratios when combined with the imputation methods in

 ISBN: 978-1-61208-352-0

TABLE 7. DIAGNOSIS OF DGA RATIOS METHOD USING *K*NN WITH DIFFERENT VALUES OF *k*

Diagnose	***k***	**DGA Diagnostic Methods**		
		Doernenburg	Rogers	IEC
Unresolved	1	34.73	**34.73**	**29.94**
	3	35.93	50.90	44.91
	5	**28.74**	51.50	44.91
	7	37.13	51.50	44.91
	10	38.32	52.10	45.50
Wrong	1	**28.14**	**13.17**	**23.95**
	3	29.34	14.37	25.15
	5	28.74	13.77	25.15
	7	29.34	13.77	25.15
	10	28.14	13.77	25.15
Correct	1	**37.13**	**52.10**	**46.11**
	3	34.73	34.73	29.94
	5	34.13	34.73	29.94
	7	33.53	34.73	29.94
	10	33.54	34.13	29.34

comparison with INCOMPLETE. But this is expected with the reduced number of unresolved diagnosis when imputed datasets were used for fault diagnosis.

D. *Case Study 2: MAL Dataset*

Table 9 shows the result of diagnoses for each *k* of the *k*NN method on the MAL dataset for each DGA diagnostic method. It is observed that when *k*=3, two out of the three ratios method (Rogers and IEC) gave the highest number of correct predictions. The number of unresolved cases were reduced the most when *k* = 3. However, higher *k* (10 for Doernenburg and Rogers, and 5 for IEC) were needed to record lower wrong diagnoses. It can be said that for this dataset, the best *k* needed to improve the performance of each ratio method varies from one ratio method to another. If the number of unresolved guesses is the most important criterion in evaluating the effectiveness of the imputation method, than *k* = 3 was the best. Table 10 shows the result of diagnoses for each ratio method on imputed datasets obtained from each imputation method compared in this study. For the *k*NN method, the result from *k*=3 was used for comparison with the other imputation methods as it reduced unresolved diagnoses the most for all ratio methods.

For this dataset, the effect of each imputation method in reducing the unresolved cases as compared to INCOMPLETE varies. While kNN, EM, MI, and REG met the aforementioned criterion for the three diagnostic method, MEAN, however, only did so for Doernenburg and IEC. For Rogers, MEAN increased the unresolved cases than INCOMPLETE. In increasing correct guesses against INCOMPLETE, only kNN, REG, and EM (albeit very slightly) were successful when paired with all of the three ratios individually. MEAN and MI had mixed results depending on the ratios used. Similar with IEC10DB dataset, *k*NN registered the highest number of correct guesses and the least number of wrong diagnoses for all of the ratio methods on this dataset. This made *k*NN registered the highest number of wrong guesses because the reduction of unresolved were significantly huge compared to other imputation methods.

E. *Analysis*

As stated above, different experiments are executed on two different datasets and the results show that:

a) imputing missing values in a DGA dataset reduce the number of unresolved diagnoses reported by the three established DGA ratios method when predicting the incipient fault in power transformer. Unresolved diagnoses are a well-known issue with the DGA ratio methods.
b) among the established imputation methods, *k*NN has the best effect to the diagnostic performances of the three DGA ratio methods on both of the datasets. The number of unresolved cases were the least and correct diagnoses were the highest for both datasets. *k*NN also outperformed INCOMPLETE in these criteria.
c) only REG and EM are as consistent as *k*NN in reducing the unresolved guesses produced by the ratios method on both dataset as well as in increasing the number of correct guesses when compared to INCOMPLETE.
d) MI, and MEAN have varied effects to the diagnostic performances of the three ratios methods depending on the datasets used. However, MEAN is the worst performer especially in increasing the number of correct guesses for both datasets for two (Rogers and IEC) ratio methods against INCOMPLETE.
e) *k*NN outperforms the other methods is expected because estimated values are calculated from observed samples having the most similar characteristics with the sample of interest (contains the missing value). *k*NN is a non-parametric method and requires no assumption with regards to pattern of missingness. *k*NN takes account correlation among data structure which coulld be the reason behind its best performance .
f) Interestingly, REG comes second to kNN. As mentioned in Section IX.B, REG is suitable for MCAR and MAR data. Using the chi-square test, both datasets are proved to be MCAR, with *p*-values <=0.001. This could be the possible reason behind the comparable performance of REG with kNN.
g) EM and MI are state-of-the art imputation methods which motivate us to use them with DGA dataset. Experimental results show that they fare behind kNN and REG. According to [16], both assume data are MAR. Because both DGA datasets are MCAR compliance, we assume this could be the reason behind the lesser quality of imputed values produced by them.
h) For both datasets, EM performs better than MI. According to [16], MI estimates better on smaller datasets than EM, but the results using DGA datasets were the opposite.
i) Doernenbug method benefits the most from the imputing procedure where the number of correct guesses increase against INCOMPLETE for most of the

experimental settings (only MEAN failed in the MAL dataset).

j) For IEC10DB, the combinations of kNN and Rogers ratio gives the highest correct guesses, whilst for MAL, they are *k*NN and IEC.

VI. CONCLUSION

The main objective of this research was to investigate the effects of missing values imputation methods on the diagnostic performances of the Doernenburg, Rogers, and IEC for predicting the incipient faults of power transformers. From the experimental results, we can safely say that, in general, imputing missing values can reduce the number of unresolved diagnoses faced by the three diagnostic methods as shown by four out five methods compared in this study. However, the number of correct guesses obtained by the ratios methods vary according to the combination of imputation method and the ratio method. Some combinations increased the correct guesses than INCOMPLETE while other combinations did the opposite. It is to be expected that the number of wrong guesses increase with the reduced number of unresolved cases. It is found that *k*NN brings the best effect to the performances of the three ratio methods compared to the EM, MI, REG, and MEAN methods. The experimental results show that imputing missing values found in a DGA dataset can bring positive effects to the performance of three established DGA diagnostic methods especially in reducing unresolved diagnoses - a common drawback faced by these diagnostic methods. We would like to undertake the study on the impact of imputing missing values to the performance of machine learning algorithms that learn from historic DGA dataset to classify fault in power transformer for future research.

TABLE 8. COMPARATIVE PERFORMANCES OF IMPUTATION METHODS ON THE DGA RATIO METHODS

Diagnose	**Imputation Methods**	**DGA Diagnostic Methods**		
		Doernenburg	Rogers	IEC
Unresolved	INCOMPLETE	45.51	56.89	50.3
	MEAN	33.53	52.69	44.91
	REG	37.13	51.94	45.51
	EM	36.53	50.30	44.31
	MI	36.53	53.29	44.31
	*k*NN	**34.73**	**34.73**	**29.94**
Wrong	INCOMPLETE	25.15	10.78	20.96
	MEAN	35.93	15.57	26.95
	REG	31.14	13.77	24.55
	EM	31.14	15.57	25.15
	MI	31.14	13.17	26.35
	*k*NN	**28.14**	**13.17**	**23.95**
Correct	INCOMPLETE	29.34	32.33	28.74
	MEAN	30.54	31.74	28.14
	REG	31.74	34.73	29.94
	EM	32.33	34.13	30.54
	MI	32.34	33.53	29.34
	*k*NN	**37.13**	**52.10**	**46.11**

TABLE 9. DIAGNOSIS OF DGA RATIOS METHOD USING *K*NN WITH DIFFERENT VALUES OF *k*

Diagnose	***k***	**DGA Diagnostic Methods**		
		Doernenburg	Rogers	IEC
Unresolved	1	58.62	83.33	81.71
	3	**6.67**	**50**	**9.11**
	5	60.73	83.17	83.41
	7	61.14	83	83.09
	10	62.68	83.58	82.85
Wrong	1	22.76	8.78	9.43
	3	76.34	31.87	50.98
	5	21.79	9.10	**9.27**
	7	21.79	9.27	9.51
	10	**20.24**	**8.62**	9.43
Correct	1	**18.62**	7.89	8.86
	3	16.99	**18.13**	**39.92**
	5	17.48	7.72	7.32
	7	17.07	7.72	7.40
	10	17.07	7.80	7.72

TABLE 10. COMPARATIVE PERFORMANCES OF IMPUTATION METHODS ON THE DGA RATIOS METHODS

Diagnose	**Imputation Methods**	**DGA Diagnostic Methods**		
		Doernenburg	Rogers	IEC
Unresolved	INCOMPLETE	84.55	90.65	89.59
	MEAN	68.70	92.60	89.51
	REG	68.78	78.54	76.67
	EM	62.68	88.86	88.04
	MI	60.96	83.58	83
	kNN	**6.67**	**50**	**9.11**
Wrong	INCOMPLETE	3.58	3.09	4.47
	MEAN	**20.98**	**2.36**	**5.69**
	REG	18.54	12.44	11.87
	EM	25.36	4.88	6.01
	MI	24.47	12.36	13.25
	kNN	76.34	31.87	50.98
Correct	INCOMPLETE	11.87	6.26	5.9
	MEAN	10.32	5.04	4.80
	REG	12.68	9.02	11.46
	EM	11.95	6.26	5.93
	MI	14.55	4.06	3.74
	kNN	**16.99**	**18.13**	**39.92**

ACKNOWLEDGMENT

The authors would like to express our appreciation to Universiti Teknikal Malaysia Melaka (UTeM) and the SLAB Programme by Ministry of Higher Education Malaysia (MOHE) for their invaluable supports either technically and financially in encouraging the authors to publish this paper.

REFERENCES

[1] Guide for the Interpretation of Gases Generated in Oil-Immersed Transformers, IEEE Std C57.104-2008 (Revision of IEEE Std C57.104-1991), 2009.

[2] IEC Publication 60599, “Mineral Oil-Impregnated Equipment in Service - Guide to the Interpretation of Dissolved and Free Gases Analysis, ” March 1999.

[3] M. Duval, “Dissolved gas analysis: It can save your transformer,” IEEE Electrical Insulation Magazine, vol. 5, Nov. 1989, pp. 22-27, doi: 10.1109/57.44605.

[4] S. W. Kim, S. J. Kim, H. D. Seo, J. R. Jung, H. J. Yang and M. Duval. (2013). “New methods of DGA diagnosis using IEC TC 10 and related databases Part 1: application of gas-ratio combinations,” IEEE Transactions on Dielectrics and Electrical Insulation, vol. 20, 2013, pp. 685-690.

[5] W. Q. Zhao, Y. L. Zhu, D. W. Wang, and X. M. Zhai, “A fault diagnosis model for power transformer based on statistical theory,” Proc. IEEE Conf. Wavelet Analysis and Pattern Recognition (ICWAPR'07), Nov. 2007 , pp. 962-966.

[6] M. Dong, D. K. Xu, M. H. Li, and Z. Yan, “Fault diagnosis model for power transformer based on statistical learning theory and dissolved gas analysis,” Proc. IEEE. Symp. Electrical Insulation, Sep. 2004, pp. 85-88.

[7] L. Zhong, Y. Jinsha, and S. Peng, “Fault Diagnosis of Power Transformer Based on Heuristic Reduction Algorithm ,” Proc. IEEE. Conf. Power and Energy Engineering (APPEEC 2009), Mar. 2009, pp. 1-4.

[8] R. Naresh, V. Sharma, and M. Vashisth, “An integrated neural fuzzy approach for fault diagnosis of transformers,” IEEE Transactions on Power Delivery, vol. 23, 2008, pp. 2017-2024.

[9] J. Liu, Y. Liang, and X. Sun, “Application of Learning Vector Quantization network in fault diagnosis of power transformer,” Proc. IEEE. Conf. Mechatronics and Automation (ICMA 2009), Aug. 2009, pp. 4435-4439.

[10] Z. Yong-li and G. Lan-qin, (2006, November). “Transformer fault diagnosis based on naive bayesian classifier and SVR,” Proc. IEEE. Conf. TENCON 2006, Nov. 2006 , pp. 1-4.

[11] B. Twala and M. Phorah, (2010). “Predicting incomplete gene microarray data with the use of supervised learning algorithms,” Pattern Recognition Letters, vol. 31, 2010, pp. 2061-2069, doi: 10.1016/j.patrec.2010.05.006.

[12] T. Yu, H. Peng, and W. Sun, (2011). “Incorporating nonlinear relationships in microarray missing value imputation,” IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 8, 2011, pp. 723-731, doi: 10.1109/tcbb.2010.73

[13] K. J. Janssen, A. R. T. Donders, F. E. Harrell Jr, Y. Vergouwe, Q. Chen, D. E. Grobbee and K. G. Moons, (2010). “Missing covariate data in medical research: to impute is better than to ignore,” Journal of clinical epidemiology, vol. 63, 2010, pp. 721-727.

[14] J. M. Jerez, I.. Molina, P. J. García-Laencina, E. Alba, N. Ribelles, M. Martín and L. Franco, “Missing data imputation using statistical and machine learning methods in a real breast cancer problem,” Artificial intelligence in Medicine, vol. 50, 2010, pp. 105-115.

[15] G. King, J. Honaker, A. Joseph and K. Scheve, “Analyzing incomplete political science data: An alternative algorithm for multiple imputation,” American Political Science Association, vol. 95, Cambridge University Press, 2001, pp. 49-69. .

[16] H. Peyre, A. Leplège and J. Coste, “Missing data methods for dealing with missing items in quality of life questionnaires. A comparison by simulation of personal mean score, full information maximum likelihood, multiple imputation, and hot deck techniques applied to the SF-36 in the French 2003 decennial health survey,” Quality of Life Research, vol. 20, 2011, pp. 287-300.

[17] J. L. Schafer and J. W. Graham, “Missing data: our view of the state of the art,” Psychological Methods, vol. 7, 2002, pp. 147-177.

[18] D. J. Stekhoven and P.Bühlmann, “MissForest—non-parametric missing value imputation for mixed-type data,” Bioinformatics, vol. 28, 2012, pp. 112-118, doi: 10.1093/bioinformatics/btr597.

[19] G. Batista and M.C. Monard, “A study of k-nearest neighbour as an imputation method,” Soft Computing Systems: Design, Management and Applications, 2002, pp. 251-260.

[20] N. Tsikriktsis, “A review of techniques for treating missing data in OM survey research,” Journal of Operations Management, vol. 24, 2005, pp. 53-62, doi: 10.1016/j.jom.2005.03.001.

[21] D.B. Rubin, “Inference and missing data,” Biometrika, vol. 63, 1976, pp. 581-592.

[22] S. Walks, “Moments and distributions of estimate of population parameters from fragments samples,” Ann. math. Stat.3, 1932, pp. 163-203.

[23] R. Little and D. Rubin, “Statistical Analysis with Missing Data,” second ed. John Wiley and Sons, New York, 2002.

[24] M. Glasser, “Linear regression analysis with missing observations among the independent variables,” Journal of the American Statistical Association 59.307,1964, pp. 834-844.

[25] A. A. Afifi and R. M. Elashoff. “Missing observations in multivariate statistics I. Review of the literature,” Journal of the American Statistical Association 61.315, 1966, pp. 595-604.

[26] C.H. Brown, “Asymptotic comparison of missing data procedures for estimating factor loadings,” Psychometrika 48.2 (1983): 269-291

[27] A. Farhangfar, L.A. Kurgan and W. Pedrycz, “A Novel Framework for Imputation of Missing Values in Databases,” IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 37, 2007, pp. 692-709.

[28] S. Zhang, Z. Jin and X. Zhu, “Missing data imputation by utilizing information within incomplete instances,” Journal of Systems and Software, vol. 84, 2011, pp. 452-459.

[29] Y.T. Mustafa, V.A. Tolpekin and A. Stein, “Application of the Expectation Maximization Algorithm to Estimate Missing Values in Gaussian Bayesian Network Modeling for Forest Growth,” IEEE Transactions on Geoscience and Remote Sensing, vol. 50, 2012, pp. 1821-1831.

[30] L. Lei, W. Naijun and L. Peng. “Applying sensitivity analysis to missing data in classifiers,” Proc. Conf. Services Systems and Services Management (ICSSSM '05), Jun. 2005, pp. 1051-1056.

[31] Q. Song, M. Shepperd, X. Chen, and J. Liu, “Can k-NN imputation improve the performance of C4.5 with small software project data sets? A comparative evaluation,” J. Syst. Softw, vol. 81, 2008, pp. 2361-2370.

[32] H. Junninen, H. Niska, K. Tuppurainen, J. Ruuskanen and M. Kolehmainen, “Methods for imputation of missing values in air quality data sets,”Atmospheric Environment, vol. 3818, 2004, pp. 2895-2907

[33] P.D. Allison, “Missing data techniques for structural equation modeling,” Journal of abnormal psychology, vol. 112, 2003, pp. 545-557.

[34] N. M. L. A. P. Dempster, D. B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm,” Journal of the Royal Statistical Society, vol. 39, 1977, pp. 1-38.

[35] R. J. A. Litte and D.B. Rubin “Statistical analysis with missing data,” New York: Wiley, 1987.

[36] E. Acuna and C. Rodriguez, “The treatment of missing values and its effect inthe classifier accuracy,” In Banks,D. et al. (eds) Classification, Clustering and DataMining Applications. Springer-Verlag, Berlin, Heidelberg, 2004, pp. 639–648.

[37] O. Troyanskaya, “Missing value estimation methods for DNA microarrays ,” Bioinformatics, vol. 17, 2001, pp. 520–525.

[38] M. Duval and A. dePabla, “Interpretation of gas-in-oil analysis using new IEC publication 60599 and IEC TC 10 databases,”, IEEE on Electrical Insulation Magazine, vol. 17, 2001, pp. 31-41.

[39] IBM Corp. Released 2012. IBM SPSS Statistics for Windows, Version 21.0. Armonk, NY: IBM Corp.

[40] MATLAB, version 7.9.0 (R2009b). 2009, The MathWorks Inc.

Unlike Behavior of Natural Frequencies in Bending Beam Vibrations with Boundary Damping in Context of Bio-inspired Sensors

Carsten Behn
and Christoph Will

Department of Technical Mechanics
Ilmenau Universiy of Technology, Germany
Email: carsten.behn@tu-ilmenau.de
Email: christoph.will@tu-ilmenau.de

Joachim Steigenberger

Institute of Mathematics
Ilmenau Universiy of Technology, Germany
Email: joachim.steigenberger@tu-ilmenau.de

***Abstract*—In this paper, we introduce certain models which arise in investigating some vibration problems of bio-inspired, vibrissa-like sensor models. Some approaches to the modeling of the biological paragon vibrissa use rigid body models in which a rod-like vibrissa is supported by a combination of spring and damping elements modeling the viscoelastic properties of the follicle-sinus complex. However, all the rigid body models can only offer limited information about the functionality of the biological sensory system. Therefore, we deal with bending problems of continuous beam systems. We present various beams with different supports (clamped and pivoted with discrete viscoelastic couplings) which are to model the biological tissues. This is new in and different from literature. We focus on investigations of the natural frequency spectra of various systems. The knowledge of dynamical characteristics is important for the design of artificial sensors. A close examination of vibrissa-like beam models with boundary damping exhibits features which are unlike in comparison to classical vibration systems.**

Keywords–Bending beam vibration; Boundary damping; Natural frequency; Bio-inspired sensor; Vibrissa.

I. INTRODUCTION

The classical Euler-Bernoulli beam is often used to analyze the vibration behavior of systems in technical disciplines like mechanical engineering, automotive engineering (e.g., power train vibration), microsystems technologies (e.g., cantilever vibration). In recent years, this classical model is used to model and to understand effects of vibrissa sensor systems in biomechanics [14]. This is the background of the work presented in the paper. Due to the biological paragon, we set up various mechanical models and analyze them in an analytical and numerical way. In contrast to works from literature [3] [17] [29], we focus on vibrissa dynamics, precisely, we try to get information about an obstacle contact in determining the spectrum of natural frequencies and calculate its shift according to an obstacle contact (sudden change of boundary conditions) [30]. In contrast to literature, we incorporate spring and damping elements, representing the biological tissue of animal skin and support of the vibrissa. This is rarely done in literature. Hence, we extend results in [21].
For this, we start an introduction to the biological paragon, describing its functionality, presenting the state of art in modeling animal vibrissae, and introduce the analytical treatment of transverse vibrations of beams due to [32] in the following.

A. Biological paragon animal vibrissa

Mice and rats use their vibrissae (in the mystacial pad) to acquire information about their surroundings. The vibrissa itself (made of dead material) is mainly used as a lever for the force transmission. But, in contrast to ordinary hairs, vibrissae are stiffer and have a (assumed hollow) conical shape [4]. The mystacial vibrissae are arranged in an array of columns and rows around the snout, see Fig. 1 and [31]. Each

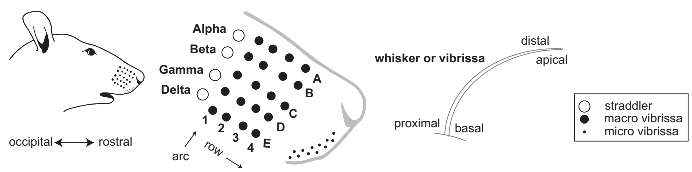


Figure 1. Schematic drawing of the mystacial pad, [31], arranged by D. Voges (TU Ilmenau).

vibrissa is embedded in and supported by its own follicle-sinus complex (FSC). The FSC is characterized by its exceptional arrangement of blood vessels, neural connections and muscles. It is presumed that the rodents can control the viscoelastic properties of the vibrissa's support by regulating the blood supply to the sinus (like a blood sac) [5]. The functionality of these vibrissae vary from animal to animal and is best developed in rodents, especially in mice and rats [16]. The detection of contact forces is made possible by the pressure-sensitive mechanoreceptors in the support of the vibrissa (i.e., FSC), see Fig. 2. These mechanoreceptors are stimulated due to the vibrissa displacements in the FSC. The nerves transmit the information through several processing units to the Central Nervous System (CNS). The receptor cells offer the fundamental principle 'adaptation'. The muscle-system, see Fig. 3 (adapted from [5] [6] [33] [12]) enables the rodents to use their vibrissae in two different ways (modes of operation): In the *passive mode*, the vibrissae are being deflected by external forces (e.g., wind). They return to their rest position passively — thus without any muscle activation, just via the fibrous band. In the *active mode*, the vibrissae are swung back- and forward by alternate contractions of the intrinsic and extrinsic muscles (with different frequencies and amplitudes). By adjusting the frequency and amplitude of the oscillations, the rodents are able to investigate object surfaces and shapes amazingly fast and with high precision [13]. *But*, how the

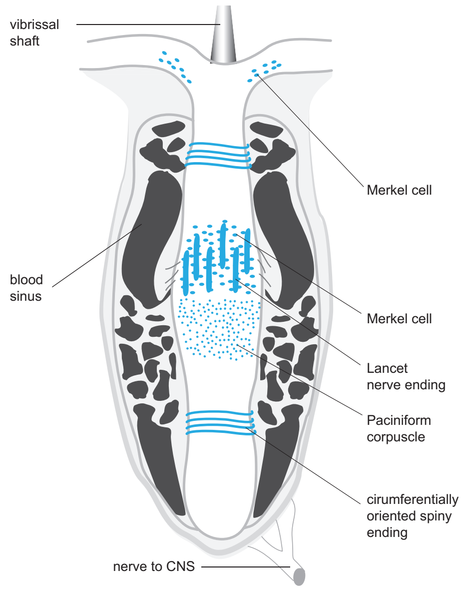


Figure 2. FSC of a vibrissa with various types of receptors (blue) [2]. Adapted from [7] [23], arranged by D. Voges (TU Ilmenau).

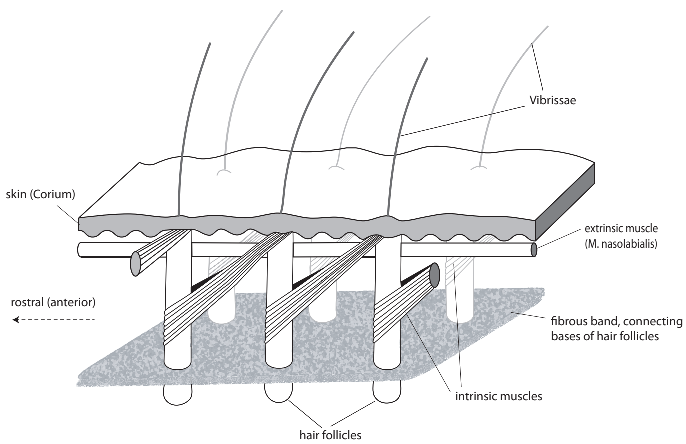


Figure 3. Schematic drawing of neighboring mystacial follicles [2]. Arranged by D. Voges (TU Ilmenau).

animals convert these multiple contacts with single objects into coherent information about their surroundings is still unclear. And it is not of main interest from our point of view: the tenor of our investigations is from bionics. The main focus is not on “copying” the solution from biology/animality, rather on detecting the main features, functionality and algorithms of the considered biological systems to implement them in (here: mechanical) models and to develop ideas for prototypes.
Therefore, this biological sensor system is highly interesting for applications in the field of autonomous robotics, since tactile sensors can offer reliable information, where conventional sensors fail (in dark, smoky or noisy environments).

B. State of art in modeling vibrissa-like sensors

Since the author in [24] tried to determine the position of a robot arm with vibrissa-like sensors (made of guitar strings), the demand for technical vibrissae grew steadily. In the meantime, these tactile sensors often complement or even replace optical sensors (as mentioned above) in their two main fields of application: flow measurements in micro technology and autonomous robotics. Especially in the latter field, technical vibrissae are currently just used to avoid collisions (merely used as contact sensors with a binary output [22]). In the last decade, the number of scientific works in which the capabilities of the tactile sensors were improved, grew significantly. As in 1996 the development of robots equipped with artificial vibrissae and driving along walls [15], was seen as a considerable achievement, the recently developed robots with a similar configuration managed to distinguish objects on the basis of their surface texture [8] [27] [34] [9] [19], or to determine form and position of nearby objects [26].
In the majority of papers found in literature, the development of innovative technical whiskers was poorly based on mechanical models of the vibrissa. In order to analyze the mechanical and especially the dynamical behavior of the vibrissa, the physical principles of the paradigm have to be identified. Therefore, abstract technical models, which describe the biological example in detail and are suitable to be analyzed using engineering and scientific methods, are sought.
Usually two types of models are used to analyze the mechanical behavior of the vibrissa:

- *Rigid body models* form the vibrissa as a stiff, inelastic body. Such models have the advantage of a simple mathematical description and solution. Furthermore, these models can easily be used to analyze the influence of varying viscoelastic supports. However, neglecting the inherent elasticity of the vibrissa implies a questionable oversimplification of the biological example.
- *Continuum models* are closer to the biological paradigm, as the tactile hair is implemented as an elastic beam. They are thus able to take the inherent dynamical behavior and the bending stiffness of the biological vibrissa into account.

An intensive literature overview of technical vibrissa models (rigid body and continuum) has been given in [2]. In the following we summarize the relevant models thereof without any valuation:

Birdwell et al. [3] - Model analyzing the bending behavior of natural vibrissae
$\oplus$ suitable to analyze the bending behavior
$\ominus$ Linearized model: only valid for small deflections
$\oplus$ Consideration of the conical shape of the vibrissa
$\ominus$ Neglecting the support’s compliance
$\oplus$ Finding: Shape of the beam influences the bending behavior
$\hookrightarrow$ not negligible
$\oplus$ Finding: Young’s modulus of natural vibrissae varies

Birdwell et al. [3] - Model to determine clamping torques
$\ominus$ Linearized model only valid for small deflections
$\oplus$ Consideration of the conical vibrissa shape
$\ominus$ Neglecting the support’s compliance
$\oplus$ Finding: influence of the natural pre-curvature of the vibrissa is negligible

Scholz and Rahn [25] - Model for profile sensing with an actuated vibrissa

⊕ Implementation of the active mode
⊖ Neglecting the support's compliance

Neimark et al. [18], Andermann et al. [1] - Model for the determination of the support's influence on the resonance properties of natural vibrissae

⊕ Experimental measurements of vibrissae's resonance frequencies
⊖ dubious results during numerical evaluations
↪ due to constant Young's modulus taken for all vibrissae
⊕ Finding: massive influence of the support on the resonance frequencies
⊖ Determination only of the first frequencies of the vibrissae
⊕ Finding: geometrically distributed sensitivity in the vibrissa array
⊕ Finding: transduction and processing of the frequency provoking stimuli to the CNS
↪ Resonance frequencies contain relevant information

There are a lot of more works concerning bending problems of vibrissa-like beams, but in context of quasi-statically object scanning and not in context of dynamical treatment, e.g., in [20].

C. Criticism and Goal of Investigations

Most of the models in literature, in particular the rigid body models, are just results of anatomic investigations. They do not directly aim at bionic applications. Further on, some models are very exact, but too complex to gain deeper insight the system to identify the essential mechanical elements.
On the other hand, in particular, concerning continuum beam models, the level of mathematical investigations is rather low:

- linear bending theory with very simple (obvious) conclusions,
- mixing of linear and nonlinear theories, and
- using boundary-value problems (BVP) which do not match the real objects sufficiently.

Based on the mentioned criticisms the **global goal** is *to present models more transparent and to use more stringent mechanics and mathematical analysis to exploit them.* The goal is **not** to recreate an exact copy of the biological system, but *to implement in a mechanical model the specific characteristics of the vibrissa needed for the detection of useful information in challenging surroundings* (**principle goal of theoretical bionics**).
A lot of works offer models consisting of beams or rigid rods for the vibrissa and mapping the arrangement of the muscles needed for the different modes of operation by viscoelastic supports. Some of those models consider a complete row of vibrissae. These models are too complex to handle and are not investigated further in those papers. Our aim is to set up simple models for the investigations first, and then to increase the complexity by adding more viscoelastic supports and to increase the degree of freedom. The viscoelastic support is very important since we have to model the compliance of the FSC and the skin, which was omitted in [18]. The boundary conditions there did not match reality, and the authors considered only the first natural frequency. We will focus on the determination of a part of the natural frequency spectrum of the vibrissa models to obtain a characteristic change depending on the change of the viscoelastic support. For these investigations we derive the equations of motion analytically to treat them with numerical tools: we try to detect useful information from the surroundings, where we focus on *changes* of environmental signals. This is quite easier to organize since the animals, more precisely, the CNS has problems in determination absolute values [3].
We point out, that we focus on a single vibrissa and not on a tuft of various vibrissae.

II. Continuum Beam Models

We will present various approaches to implement and to determine the basic features of animal vibrissae as mentioned in Subsection I-A.
Here, we will focus on the mechanical properties and the dynamic behavior of the vibrissa beam models. The processing of the stimulus and the corresponding analysis of different control strategies are **not** discussed here. Furthermore, the investigations are addressed to a single vibrissa – the interaction between the different vibrissae in the mystacial pad is **not** taken into account.

The classical differential equation for small bending vibrations of beams (linear Euler-Bernoulli theory) is the basis of the investigations. We will set up and analyze various vibrissa beam models with different supports using discrete and continuously distributed spring and damping elements to mimic tissues of FSC and skin. Following [18], we focus on the determination of the natural frequency spectrum of such beams analytically and numerically, while varying the viscoelastic properties of the support. We will not focus on static bending problems in the following.

Starting point and motivation of the following investigations are multiple **hypotheses** concerning the **functionality of the vibrissa**:

- The elasticity and the conical shape of the hair are relevant for the functionality of the vibrissa [3].
- The viscoelastic properties of the support (see the FSC) are controlled by the blood pressure in the blood sinus [5] [4].
- The vibrissae are excited with or close to their resonance frequencies during the active mode [18] [1].

Following these hypotheses, the primary tasks now are:

- to investigate the influence of elasticity and conical shape on the vibration characteristics of the vibrissa by analyzing its natural frequency spectrum;
- to analytically examine innovative models of a flexible vibrissa with a viscoelastic support which fit the real object and its support better than models in literature.

A. Introduction to Transversal Bending Beam Vibrations

Let us start with the following example: a one-sided clamped beam with elastic support (spring stiffness c) at the

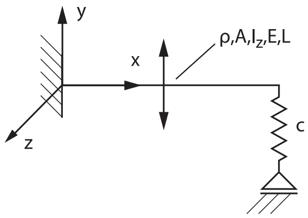


Figure 4. One-sided clamped beam with elastic end support.

end, see Fig. 4. The beam has length L, Young's modulus E, density ϱ, constant cross section area A and second moment of area I_z. We are seeking for the first five *natural frequencies*.

Remark II.1. *We focus on the first five natural frequencies of the spectrum because of*

1. *mathematical reasons: the first five natural frequencies will form a good approximation basis of the Fourier series of the solution made by the method of separation of variables; and*
2. *physical meanings – higher natural frequencies are too large, whereas only lower ones are perceptible by means of tactile sense.*

The well-known equation of motion for free vibrations of a beam with small deformations, as in Fig. 4, is:

$$\ddot{v}(x,t) + k^4\, v^{''''}(x,t) = 0\,, \quad \text{with} \quad k^4 := \frac{E\,I_z}{\varrho\, A}\,, \tag{1}$$

where the function $v(x,t)$ describes the vertical displacement at point x and at time t.
The partial differential equation (PDE) (1) and the following boundary conditions
① : $v(0,t) = 0 \;\forall\, t \geq 0$
② : $v'(0,t) = 0 \;\forall\, t \geq 0$
③ : $v''(L,t) = 0 \;\forall\, t \geq 0$
④ : $v'''(L,t)\, E\, I_z - c\, v(L,t) = 0 \;\forall\, t \geq 0$
form a BVP.

Now, we apply the *method of separation of variables*, i.e., we are seeking for special solutions of structure

$$v(x,t) = X(x) \cdot T(t) \quad \forall\, (x,t)\,. \tag{2}$$

Substitution into (1) yields two ordinary differential equations (ODEs)

$$\frac{\ddot{T}(t)}{T(t)} = -\mu^2\,, \tag{3}$$

$$-k^4\, \frac{X''''(x)}{X(x)} = -\mu^2\,. \tag{4}$$

The general solution of 3 is

$$T(t) = \underline{B}_1\, e^{i\,\mu\,t} + \underline{B}_2\, e^{-i\,\mu\,t}\,,\; \underline{B}_1, \underline{B}_2 \in \mathbb{C}\,. \tag{5}$$

The solution of 4 is:

$$X(x) = C_1\, \cos(\lambda\, x) + C_2\, \sin(\lambda\, x) + C_3\, \cosh(\lambda\, x) + C_4\, \sinh(\lambda\, x)\,. \tag{6}$$

with $C_1, C_2, C_3, C_4 \in \mathbb{C}$ and

$$\boxed{\lambda^4 := \frac{\mu^2}{k^4}\,, \qquad k^4 := \frac{E\,I_z}{\varrho\,A}\,.} \tag{7}$$

This shape solution (6) together with the formulated four boundary conditions form an eigenvalue problem (EVP) in the following. We get $\forall\, t \geq 0$

① $T(t)\;(C_1 + C_3) = 0$

② $T(t)\,\lambda\;(C_2 + C_4) = 0$

③ $T(t)\,\lambda^2\,\big(-C_1\,\cos(\lambda\,L) - C_2\,\sin(\lambda\,L) + C_3\,\cosh(\lambda\,L) + C_4\,\sinh(\lambda\,L)\big) = 0$

④ $E\,I_z\,T(t)\,\lambda^3\,\big(C_1\,\sin(\lambda\,L) - C_2\,\cos(\lambda\,L) + C_3\,\sinh(\lambda\,L) + C_4\,\cosh(\lambda\,L)\big) - c\,T(t)\,\big(C_1\,\cos(\lambda\,L) + C_2\,\sin(\lambda\,L) + C_3\,\cosh(\lambda\,L) + C_4\,\sinh(\lambda\,L)\big) = 0$

$T(t)$ drops, and a system of homogenous linear equations results with a coefficient matrix (8).

Since we are seeking for non-trivial solutions, we claim the singularity of the coefficient matrix: $\det(M) = 0$. Introducing a ratio of elasticity

$$\gamma_c := \frac{c}{c_S} = \frac{c}{\frac{E\,I_z}{L^3}} = \frac{c\,L^3}{E\,I_z}$$

we obtain the characteristic eigenvalue equation

$$\lambda^3\,L^3\;(1 + \cosh(\lambda\,L)\,\cos(\lambda\,L)) + \gamma_c\;(\cosh(\lambda\,L)\,\sin(\lambda\,L) - \cos(\lambda\,L)\,\sinh(\lambda\,L)) = 0 \tag{9}$$

Remark II.2. *Before solving* (9) *we check it in setting*

- $c = 0$*: we get* $1 + \cosh(\lambda\,L)\cos(\lambda\,L) = 0$*, which forms the eigenvalue equation of an one-sided clamped / free end beam;*
- $c \to +\infty$*: we get* $\cosh(\lambda\,L)\sin(\lambda\,L) - \cos(\lambda\,L)\sinh(\lambda\,L) = 0$*, which arises for a clamped beam with bearing.*

Now, we present some numerical calculations. We are varying $\gamma_c = 0,\, 0.1,\, 1,\, +\infty$ and derive the natural frequencies of a steel beam and of a $B2$ vibrissa, see Fig. 1, using the following parameters:

- steel beam: $E = 210\,\text{GPa}$, $\varrho = 7850\,\frac{\text{kg}}{\text{m}^3}$;
- $B2$ vibrissa: $E = 2.3\,\text{GPa}$, $\varrho = 238.732\,\frac{\text{kg}}{\text{m}^3}$;
- geometric parameters: $d = 0.2\,\text{mm}$, $I_z = \frac{\pi}{64}\,d^4$, $A = \frac{\pi}{4}\,d^2$, $L = 40\,\text{mm}$.

The following tables present the first five eigenvalues λ_j, natural frequencies ω_j in rad/s and frequencies f_j in Hz for a steel beam and a $B2$ vibrissa.

 ISBN: 978-1-61208-352-0

$$M(\lambda) := \begin{pmatrix} 1 & \vdots & 0 & \vdots & 1 & \vdots & 0 \\ \hdashline 0 & \vdots & \lambda & \vdots & 0 & \vdots & \lambda \\ \hdashline \cos(\lambda L)\,\lambda^2 & \vdots & \sin(\lambda L)\,\lambda^2 & \vdots & \cosh(\lambda L)\,\lambda^2 & \vdots & \sinh(\lambda L)\,\lambda^2 \\ \hdashline E\,I_z\,\sin(\lambda L)\,\lambda^3 & \vdots & -E\,I_z\,\cos(\lambda L)\,\lambda^3 & \vdots & E\,I_z\,\sinh(\lambda L)\,\lambda^3 & \vdots & E\,I_z\,\cosh(\lambda L)\,\lambda^3 \\ -c\,\cos(\lambda L) & \vdots & -c\,\sin(\lambda L) & \vdots & -c\,\cosh(\lambda L) & \vdots & -c\,\sinh(\lambda L) \end{pmatrix} \tag{8}$$

TABLE I. CALCULATION FOR $\gamma_c = 0$.

		steel beam		$B2$ vibrissa	
j	λ_j	ω_j	f_j	ω_j	f_j
1	$1.875\,\frac{1}{L}$	568.297	90.447	733.807	54.279
2	$4.694\,\frac{1}{L}$	3561.458	566.824	4598.693	340.159
3	$7.855\,\frac{1}{L}$	9972.187	1587.123	12876.473	952.454
4	$10.996\,\frac{1}{L}$	19541.506	3110.127	25232.748	1866.429
5	$14.137\,\frac{1}{L}$	32303.509	5141.263	41711.541	3085.341

TABLE II. CALCULATION FOR $\gamma_c = 1$.

		steel beam		$B2$ vibrissa	
j	λ_j	ω_j	f_j	ω_j	f_j
1	$2.010\,\frac{1}{L}$	653.008	103.929	843.189	62.369
2	$4.704\,\frac{1}{L}$	3576.197	569.169	4617.724	341.566
3	$7.857\,\frac{1}{L}$	9977.433	1587.958	12883.248	952.955
4	$10.996\,\frac{1}{L}$	19544.181	3110.553	25236.203	1866.685
5	$14.138\,\frac{1}{L}$	32305.127	5141.521	41713.630	3085.496

TABLE III. CALCULATION FOR $\gamma_c = +\infty$.

		steel beam		$B2$ vibrissa	
j	λ_j	ω_j	f_j	ω_j	f_j
1	$3.927\,\frac{1}{L}$	2492.061	396.624	3217.846	238.019
2	$7.069\,\frac{1}{L}$	8075.874	1285.315	10427.881	771.335
3	$10.210\,\frac{1}{L}$	16849.666	2681.708	21756.941	1609.329
4	$13.352\,\frac{1}{L}$	28813.927	4585.879	37205.657	2752.048
5	$16.493\,\frac{1}{L}$	43968.656	6997.829	56774.030	4199.492

Increasing γ_c leads to increasing ω_j, see Table I to III. Let us further point out, that these first investigations of a simple beam model are rather obvious. In the following we will increase the level of complexity.

B. Bending Beam Vibrations with vibrissa-like Support

Here, we focus on various supports (no clamps) of the vibrissa beam model. In order to do the following investigations analytically, we neglect the conical shape of the vibrissa with respect to the complex structure of the arising PDE. We focus on *cylindrical* beams.
First vibrissa beam models are presented in Figs. 5 and 6. These models present a cylindrical pivoted beam with various elastic couplings. The analytical investigations are carried out in formulating the boundary value problems (BVPs) for each section of the beam. The arising eigenvalue problems could be treated analytically in parts.
But, all models offer the same drawback: the 'pivot' is the base of the vibrissa, this does not match the reality. Therefore, we modify these models: first we shifted the pivot, and second we added some viscous properties to the support. This results in the following models, shown in Figs. 7 and 8 The BVPs of the oscillating problems are formulated in the following:

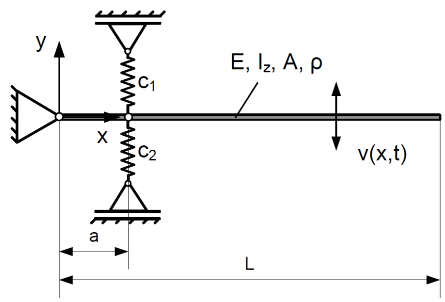


Figure 5. Pivoted vibrissa beam model with modeled skin support (one level of elasticity), [2].

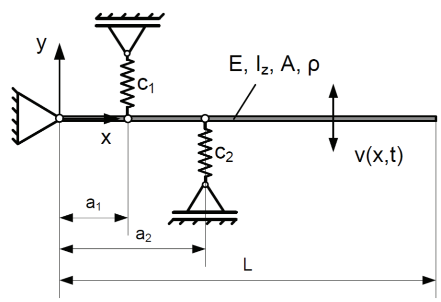


Figure 6. Pivoted vibrissa beam model with two levels of elasticity (FSC and skin), [2].

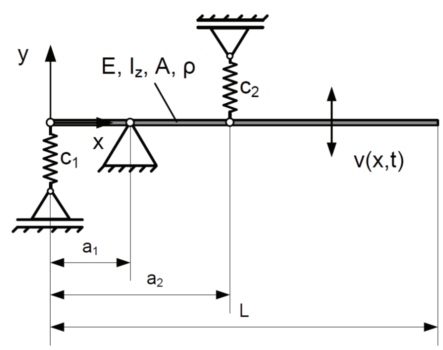


Figure 7. Undamped vibrissa beam model with modeled skin and FSC support, [2].

- undamped model in Fig. 7:
  PDEs: $\ddot{v}_i(x,t) + k^4\,v_i''''(x,t) = 0$, with $k^4 := \frac{E\,I_z}{\varrho\,A}$, $i = 1, 2, 3$,

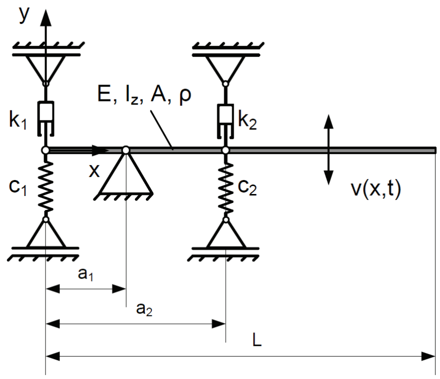


Figure 8. Damped vibrissa beam model with modeled skin and FSC support, [2].

boundary conditions:

$$
\begin{aligned}
&v_1''(0,t) = 0\,,\\
&-E\,I_z\,v_1'''(0,t) - c_1\,v_1(0,t) = 0\,,\\
&v_1(a_1,t) = v_2(a_1,t)\,,\\
&v_1(a_1,t) = 0\,,\\
&v_1'(a_1,t) = v_2'(a_1,t)\,,\\
&v_1''(a_1,t) = v_2''(a_1,t)\,,\\
&v_2(a_2,t) = v_3(a_2,t)\,,\\
&v_2'(a_2,t) = v_3'(a_2,t)\,,\\
&v_2''(a_2,t) = v_3''(a_2,t)\,,\\
&E\,I_z\,\big(v_2'''(a_2,t) - v_3'''(a_2,t)\big) - c_2\,v_2(a_2,t) = 0\,,\\
&v_3''(L,t) = 0\,,\\
&v_3'''(L,t) = 0
\end{aligned}
$$

- damped model in Fig. 8:
  PDEs: $\ddot{v}_i(x,t) + k^4\,v_i''''(x,t) = 0$, with $k^4 := \frac{E\,I_z}{\varrho\,A}$, $i = 1, 2, 3$,
  boundary conditions:

$$
\begin{aligned}
&v_1''(0,t) = 0\,,\\
&-E\,I_z\,v_1'''(0,t) - c_1\,v_1(0,t) - k_1\,\dot{v}_1(0,t) = 0\,,\\
&v_1(a_1,t) = v_2(a_1,t)\,,\\
&v_1(a_1,t) = 0\,,\\
&v_1'(a_1,t) = v_2'(a_1,t)\,,\\
&v_1''(a_1,t) = v_2''(a_1,t)\,,\\
&v_2(a_2,t) = v_3(a_2,t)\,,\\
&v_2'(a_2,t) = v_3'(a_2,t)\,,\\
&v_2''(a_2,t) = v_3''(a_2,t)\,,\\
&E\,I_z\,\big(v_2'''(a_2,t) - v_3'''(a_2,t)\big) - c_2\,v_2(a_2,t)\\
&\qquad\qquad\qquad\qquad - k_2\,\dot{v}_2(a_2,t) = 0\,,\\
&v_3''(L,t) = 0\,,\\
&v_3'''(L,t) = 0\,,
\end{aligned}
$$

To investigate the dependence of the natural frequencies on the system parameters, the eigenvalue problems (EVPs) are derived analytically (linear equations with zero determinant) and solved numerically for various cases. The following two examples illustrate some results.

Example II.3. *Let us remind the comparison of a steel beam and a $B2$ vibrissa. Due to some techniques we are able to handle discrete damping terms to analytically derive the EVP, which then can be solved numerically. There are no problems in case of small damping coefficients, due to the biological paradigm.*
We set

- *the geometric parameters* $a_1 = 3\,\mathrm{mm}$, $a_2 = 4\,\mathrm{mm}$, $r = 0.1\,\mathrm{mm}$, *and* $L = 40\,\mathrm{mm}$;
- *the support parameters for the FSC* $c_1 = c_{\mathrm{FSC}} = 80\,\frac{\mathrm{N}}{\mathrm{m}}$ *and* $k_1 = d_{\mathrm{FSC}} = 0.5\,\frac{\mathrm{Ns}}{\mathrm{m}}$;
- *and for the skin* $c_2 = c_{\mathrm{skin}} = 5.7\,\frac{\mathrm{N}}{\mathrm{m}}$ *and* $k_2 = d_{\mathrm{skin}} = 0.2\,\frac{\mathrm{Ns}}{\mathrm{m}}$.

We get the results in Tables IV and V, where we present the first five eigenvalues λ_j, the first five natural frequencies ω_j in rad/s, *and the decay rate δ_j in* 1/s *for both undamped and damped*

- *steel beam: parameters* $E = 210\,\mathrm{GPa}$ *and* $\varrho = 7850\,\frac{\mathrm{kg}}{\mathrm{m}^3}$, *and*
- *$B2$ vibrissa: parameters:* $E = 2.3\,\mathrm{GPa}$ *and* $\varrho = 238.732\,\frac{\mathrm{kg}}{\mathrm{m}^3}$.

TABLE IV. CALCULATION FOR THE STEEL BEAM.

	undamped		damped		
j	λ_j	ω_j	λ_j	ω_j	δ_j
1	3.946	2517.314	$2.017 - 0.271\,I$	645.851	176.755
2	7.448	8965.159	$4.941 - 0.085\,I$	3944.771	135.721
3	10.800	18852.940	$8.297 - 0.005\,I$	11126.480	129.665
4	14.004	31698.644	$11.650 + 0.068\,I$	21936.355	257.106
5	16.934	46348.499	$15.023 + 0.150\,I$	36477.396	727.763

TABLE V. CALCULATION FOR THE B2 VIBRISSA.

	undamped		damped		
j	λ_j	ω_j	λ_j	ω_j	δ_j
1	1.990	384.836	$1.993 + 0.033\,I$	385.694	12.594
2	4.988	2416.865	$5.151 + 0.067\,I$	2577.281	66.675
3	8.354	6779.153	$8.651 + 0.047\,I$	7270.715	78.835
4	11.703	13306.010	$12.119 + 0.037\,I$	14267.203	87.093
5	15.058	22027.360	$15.586 + 0.031\,I$	23599.899	95.129

Considering the steel beam, the (natural) frequencies shrink if we focus on a damped system, as expected. But, we observe (see Table V) an unlike behavior simulating the $B2$ vibrissa as the (natural) frequencies increase in the damped system. This contradicts the classical assertions. The reason for this is a little bit unclear, we shall have a closer look to the modes of the beams.
Further, we hint to some problems in using discrete damping elements in the next Subsection II-C.

Short summary:
⊖ Neglecting the conical shape of the vibrissa
⊕ Consideration of the support's compliance
· at skin level
· at the level of the FSC
⊕ Finding: massive influence of the support on the natural frequencies
⊕ Finding: influence of damping elements in the support
↪ massive for the 1^{st} natural frequency
↪ but: unlike behavior of the natural frequencies

C. Beam with discrete damping elements

To clarify the unlike effects of the foregoing subsection, we deal with a 'simple' problem to investigate the influence of discrete damping elements. We consider a cylindrical, one-sided clamped beam which is viscoelastically supported at the end, see Fig. 9. The well-known PDE from the linear

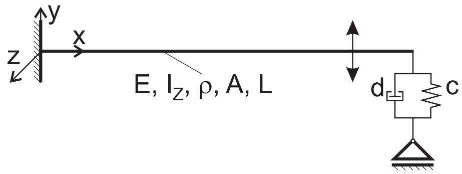


Figure 9. Clamped beam with viscoelastic end support.

Euler-Bernoulli theory is (1), which forms with the boundary conditions

$$\begin{aligned} v(0,t) &\equiv 0 \\ v'(0,t) &\equiv 0 \\ v''(L,t) &\equiv 0 \\ E\,I_z\,v'''(L,t) - d\,\dot{v}(L,t) - c\,v(L,t) &\equiv 0\,, \end{aligned}$$

a BVP.
The handling of the last boundary condition results in

$$E\,I_z\,X'''(L) - c\,X(L) = \pm\,i\,d\,\lambda^2\,k^2\,X(L)\,.$$

All conditions lead to the coefficient matrix (10) of the homogenous systems whose singularity yields the eigenvalue equation:

$$\begin{aligned} \det\left(A(\lambda)\right) = &-E\,I_z\,\lambda^3 \\ &- E\,I_z\,\cos\left(\lambda\,L\right)\,\cosh\left(\lambda\,L\right)\,\lambda^3 \\ &\pm\,i\,d\,k^2\,\sin\left(\lambda\,L\right)\,\cosh\left(\lambda\,L\right)\,\lambda^2 \\ &- c\,\sin\left(\lambda\,L\right)\,\cosh\left(\lambda\,L\right) \\ &\mp\,i\,d\,k^2\,\cos\left(\lambda\,L\right)\,\sinh\left(\lambda\,L\right)\,\lambda^2 \\ &+ c\,\cos\left(\lambda\,L\right)\,\sinh\left(\lambda\,L\right) = 0\,. \end{aligned} \tag{11}$$

Remark II.4. *At this stage, we could check this equation in concluding well-known eigenvalue equations: setting* $\{d = 0, c = 0\}$*, or* $\{d = 0, c > 0\}$*, or* $\{d = 0, c \to +\infty\}$ *results in the equations presented in [10] or [32].*

Introducing the dimensionless parameters

$$\begin{aligned} \alpha_c &:= \frac{c}{\frac{E I_z}{L^3}} \\ \alpha_d &:= \frac{L\,d}{\sqrt{\varrho\,A\,E\,I_z}}\,, \end{aligned}$$

we determine the first three natural frequencies in varying α_c and α_d. We get the following Figs. 10 to 12.

For fixed α_c and varying α_d, there are parameter ranges of α_c where we get an expected and unexpected behavior of the first natural frequency, see Fig. 10:

- $\alpha_c \in [0, 17]$: the natural frequency breaks down to zero for increasing α_d;
- $\alpha_c \in [18, 23]$: first, the natural frequency increases and then breaks down to zero;
- $\alpha_c > 23$: the natural frequency just increases.

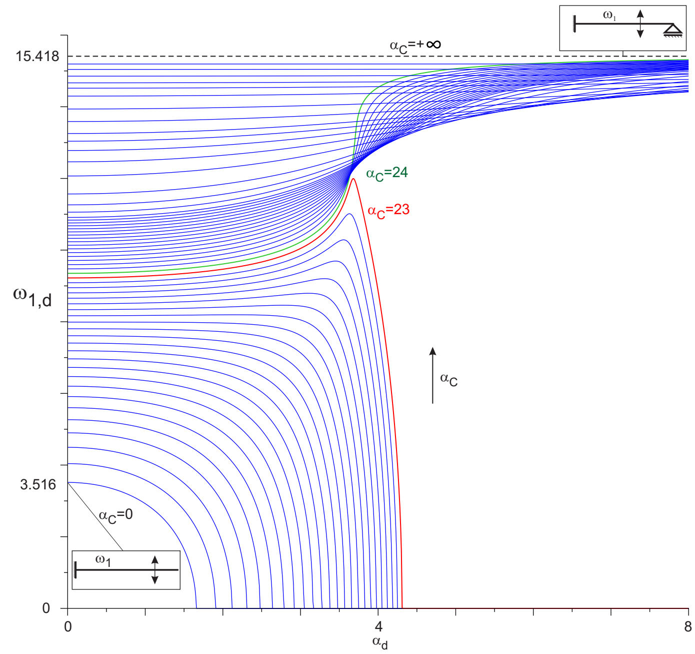


Figure 10. First natural frequency of the beam model $\omega_{1,d}$ in $\sqrt{\frac{E\,I_z}{\varrho\,A\,L^4}}$ in dependence on α_c and α_d.

On the other hand, for fixed α_d and varying α_c, we observe the following:

- $\alpha_d \in [0, 3.5]$: increasing α_c leads to an increase of the natural frequency;
- $\alpha_d > 3.5$: an increase of α_c results first in a decrease and then in an increase of the natural frequency.

This may explain the behaviors of the natural frequencies in Example II.3.
Similar effects can be observed in Figs. 11 and 12.

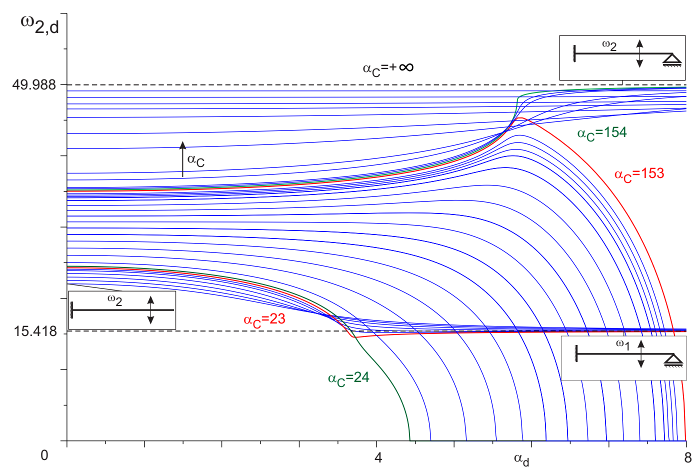


Figure 11. Second natural frequency of the beam model $\omega_{2,d}$ in $\sqrt{\frac{E\,I_z}{\varrho\,A\,L^4}}$ in dependence on α_c and α_d.

It seems that some bifurcation is happening there. This has to be checked in future.
Former investigations on similar beam models are done in

$$A(\lambda) := \left(\begin{array}{c:c:c:c} 1 & 0 & 1 & 0 \\ \hdashline 0 & \lambda & 0 & \lambda \\ \hdashline EI_z \sin(\lambda L)\, \lambda^2 & -EI_z \cos(\lambda L)\, \lambda^2 & EI_z \sinh(\lambda L)\, \lambda^2 & EI_z \cosh(\lambda L)\, \lambda^2 \\ \pm \mathrm{i}\, d\lambda^2 k^2 \cos(\lambda L) & \pm \mathrm{i}\, d\lambda^2 k^2 \sin(\lambda L) & \pm \mathrm{i}\, d\lambda^2 k^2 \cosh(\lambda L) & \pm \mathrm{i}\, d\lambda^2 k^2 \sinh(\lambda L) \\ -c \cos(\lambda L) & -c \sin(\lambda L) & -c \cosh(\lambda L) & -c \sinh(\lambda L) \\ \hdashline -\cos(\lambda L)\, \lambda^2 & -\sin(\lambda L)\, \lambda^2 & \cosh(\lambda L)\, \lambda^2 & \sinh(\lambda L)\, \lambda^2 \end{array} \right) \tag{10}$$

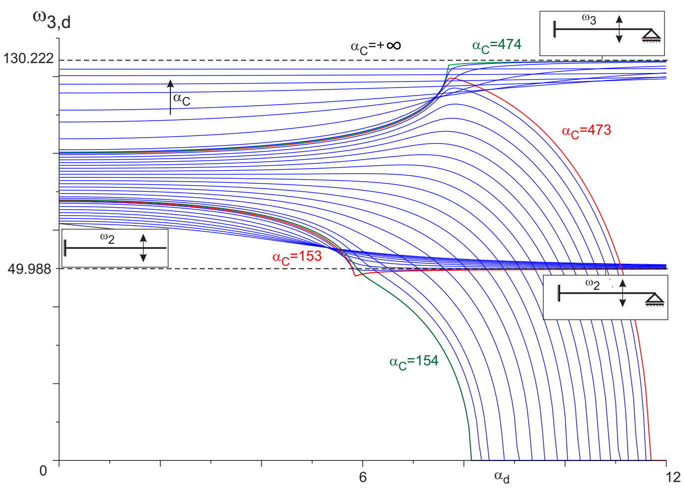


Figure 12. Third natural frequency of the beam model $\omega_{3,d}$ in $\sqrt{\frac{E\,I_z}{\varrho\,A\,L^4}}$ in dependence on α_c and α_d.

[11], with focus on eigenvalues, and with focus on vibration amplitudes. But, no one derived the above curves of the behavior of natural frequencies.

III. Conclusion

The goal of this contribution was to present the theoretical context needed to examine the mechanical and in particular the dynamical characteristics of the biological vibrissa. Moreover, these theoretical aspects were to be interpreted with respect to the biological vibrissa, as well as for a technical implementation of it. Inspired by this biological sensory system, several types of mechanical models were developed based on findings in the literature.

The second focus was on the modeling of the vibrissa as a *continuous system*: bending vibrations of beams. There, the main focus of the studies lay on the examination of the influence of the tactile hair compliance and the viscoelastic support on the oscillation characteristics of the vibrissa. The conical form was neglected until now.
The influence of the viscoelastic support of the vibrissa has been examined using various abstract models in which the vibrissa was modeled as a thin, cylindrical, flexible beam. The viscoelastic properties of the FSC and the skin were implemented by using spring and damping elements.

The damping element significantly increased the complexity of the differential equations and led to a surprising phenomenon: there exist some natural frequencies which break down to zero for a certain range of parameters. This fact is well-known in 1-DoF systems (i.e., strong damping, creeping behavior). The study demonstrated that the oscillation behavior of an elastic beam differs remarkably from the behavior of such a classical system:

- The natural frequencies may increase with growing boundary damping.
- For specific damping parameter values, the natural frequencies grow for decreasing boundary stiffness.

Some similar effects on and the behavior of the natural frequencies can be observed in analyzing the model presented in Fig. 13. For a fixed parameter set of the system, except the

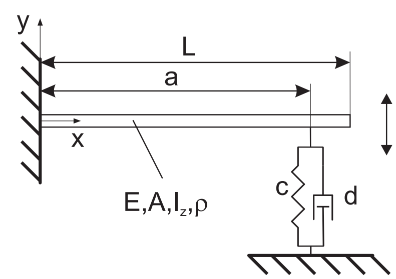


Figure 13. One-sided clamped beam with viscoelastic support.

distance a of the viscoelastic support to the clamping, we get the following results of the first three natural frequency of the beam, see Fig. 14 to 16, which offer already the same unlike behavior as the example above.

But, theories gained from the simplified linear Euler-Bernoulli theory are only valid for small deflections and deformations. If one considers a vibrissa beam in passive mode, then it may be questionable if this theory is really qualified for the investigations, see large bending deformations. Inspecting these vibrissa configurations, one could clearly observe that the vibrissa in passive mode suffers large deformations. Hence, the linear Euler-Bernoulli theory is not qualified to determine the natural frequencies since it describes the bending behavior for small deformations. We have to turn to a nonlinear theory: Timoshenko theory or nonlinear Euler-Bernoulli theory. We will arrive at more realistic models and description of these models, which then are closer to the biological paradigm. An

 ISBN: 978-1-61208-352-0

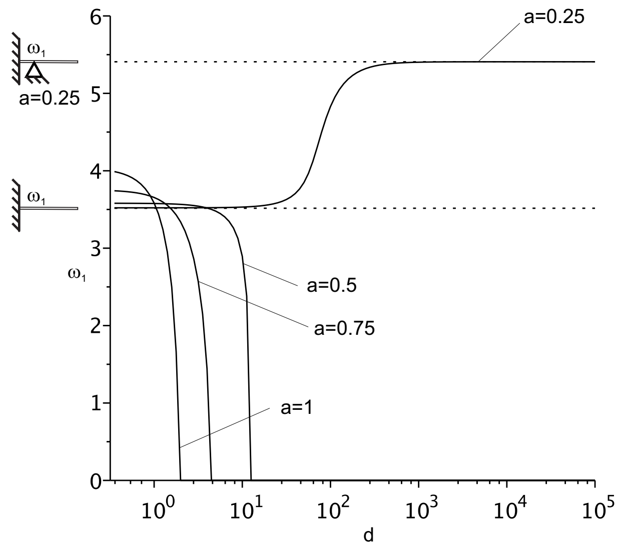


Figure 14. The first natural frequency vs. damping rate for fixed spring rate.

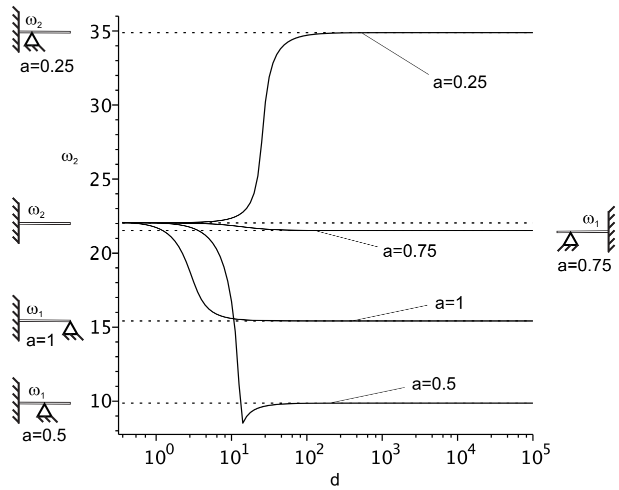


Figure 15. The second natural frequency vs. damping rate for fixed spring rate.

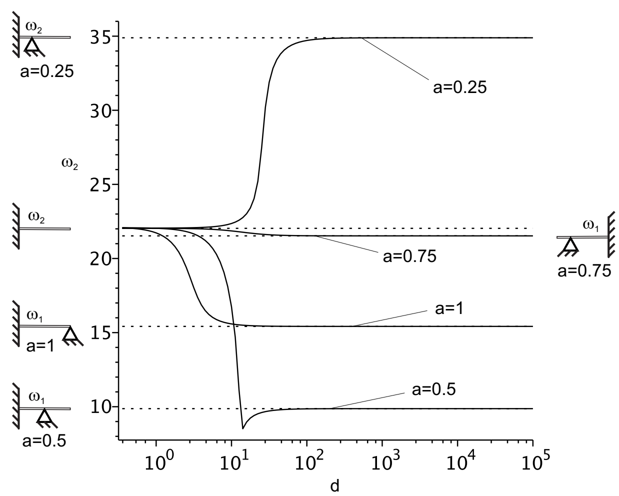


Figure 16. The third natural frequency vs. damping rate for fixed spring rate.

approach is done in [28].
However, we are focussing on long, slender beams, whereby shear forces may have less influence. So, we shall focus on the nonlinear Euler-Bernoulli theory in future work. Additionally, we shall include the conical shape and a precurvature of the beam, neglected until now.

REFERENCES

[1] M. L. Andermann, J. Ritt, M. Neimark, and C. I. Moore, Neural Correlates of Vibrissa Resonance: Band-Pass and Somatotopic Representation of High-Frequency Stimuli. Neuron 42 (2004), pp. 451–463.

[2] C. Behn, Mathematical Modeling and Control of Biologically Inspired Uncertain Motion Systems with Adaptive Features. Habilitation thesis, Faculty of Mechanical Engineering, Ilmenau University of Ilmenau, Germany (2013).

[3] A. Birdwell et al., Biomechanical Models for Radial Distance Determination by the Rat Vibrissal System. The Journal of Neurophysiology 98 (2007), pp. 2439–2455.

[4] K. Carl, Technische Biologie des Tasthaar-Sinnessystems als Gestaltungsgrundlage für taktile stiftführende Mechanosensoren, Technical biology of the vibrissa sensor system as design principles for tactile mechanosensors. PhD thesis, Universitätsverlag, Ilmenau (2009).

[5] J. Dörfl, The musculature of the mystacial vibrissae of the white mouse. Journal of Anatomy 135 (1982), pp. 147–154.

[6] J. Dörfl, The innervation of the mystacial region of the white mouse – A topographical study. Journal of Anatomy 142 (1985), pp. 173–184.

[7] S. Ebara, K. Kumamoto, T. Matsuura, J. E. Mazurkiewicz, and F. L. Rice, Similarities and Differences in the Innervation of Mystacial Vibrissal Follicle-Sinus Complexes in the Rat and Cat: A Confocal Microscopic Study. The Journal of Comparative Neurology 449 (2002), pp. 103–119.

[8] M. Fend, S. Bovet, and H. Yokoi, An Active Artificial Whisker Array for Texture Discrimination. In: Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (2003), pp. 1044–1049.

[9] M. Fend, Whisker-Based Texture Discrimination on a Mobile Robot. In: Proceedings of the 8th European Conference on Artificial Life (ECAL) (2005), pp. 302–311.

[10] D. Gross, W. Hauger, W. Schnell, and P. Wriggers, Technische Mechanik: Band 4 Hydromechanik, Elemente der Höheren Mechanik, Numerische Methoden, Technical Mechanics: Volume 4 Hydromechanics, higher mechanics, numerical methods. Springer, Berlin (2002).

[11] M. Gürgöze and H. Erol, Determination of the frequency response function of a cantilevered beam simply supported in-span. Journal of Sound and Vibration 247(2) (2001), pp. 372–378.

[12] S. Haidarliu, E. Simony, D. Golomb, and E. Ahissar, Muscle Architecture in the Mystacial Pad of the Rat. The Anatomical Record 293 (2010), pp. 1192–1206.

[13] M.J. Hartmann and J.H. Solomon, Robotic whiskers used to sense features: Whiskers mimicking those of seals or rats might be useful for underwater tracking or tactile exploration. NATURE 443 (2006), p. 525.

[14] S. Hirose, S. Inoue, and K. Yoneda, The whisker sensor and the transmission of multiple sensor signals. Advanced Robotics 4(2) (1989), pp 105-117.

[15] D. Jung and A. Zelinsky, Whisker based mobile robot navigation. In: Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems '96 (IROS 96) (1996), pp. 497–504.

[16] T.-E. Jin, V. Witzemann, and M. Brecht, Fiber Types of the Intrinsic Whisker Muscle and Whisking Behavior. The Journal of Neuroscience 24(13) (2004), pp. 3386–3393.

[17] D. Kim and R. Möller, Biomimetic whiskers for shape recognition. Robotics and Autonomous Systems 55(3) (2007), pp. 229-243.

[18] M.A. Neimark, M.L. Andermann, J.J. Hopfield, and C.I. Moore, Vibrissa Resonance as a Transduction Mechanism for Tactile Encoding. The Journal of Neuroscience 23(16) (2003), pp. 6499–6509.

[19] S. N'Guyen, P. Pirim, and J.-A. Meyer, Tactile texture discrimination in the robot-rat "Psikharpax". In: BIOSIGNALS 2010 – Proceedings of 3rd Int. Conf. on Bio-Inspired Systems and Signal Processing (2010), pp. 74–81.

[20] L. Pammer et al., The mechanical variables underlying object localization along the axis of the whisker. The Journal of Neuroscience: the Official Journal of the Society for Neuroscience 33(16) (2013), pp. 6726–6741.

[21] H. Pierson, J. Brevick, and K. Hubbard, The effect of discrete viscous damping on the transverse vibration of beams. Journal of Sound and Vibration 332 (2013), pp. 4045–4053.

[22] T. Prescott, M. Pearson, B. Mitchinson, J. Sullivan, and A. Pipe, Whisking with Robots - From Rat Vibrissae to Biomimetic Technology for Active Touch. IEEE Robotics and Automation Magazine 16(3) (2009, pp. 42–50.

[23] F.L. Rice, A. Mance, and B.L. Munher, A comparative light microscopic analysis of the sensory innervation of the mystacial pad. I. Innervation of vibrissal follicle-sinus complexes. The Journal of Comparative Neurology 252 (1986), pp. 154–174.

[24] R. A. Russell, Closing the sensor-computer-robot control loop. Robotics Age 6 (1984), pp. 15–20.

[25] G. Scholz and C. Rahn, Profile Sensing With an Actuated Whisker. IEEE Transactions on Robotics and Automation 20 (2004), pp. 124–127.

[26] A. E. Schultz, J. H. Solomon, M. A. Peshkin, and M. J. Z. Hartmann, Multifunctional Whisker Arrays for Distance Detection, Terrain Mapping, and Object Feature Extraction. In: Proceedings of 2005 IEEE International Conference on Robotics & Automation (2005), pp. 2588–2593.

[27] A. K. Seth, J. L. McKinstry, G. M. Edelman, and J. L. Krichmar, Texture discrimination by an autonomous mobile brain-based device with whiskers. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA ’04), San Diego, USA (2004), pp. 4925–2930.

[28] J. H. Solomon and M. J. Z. Hartmann, Extracting object contours with the sweep of a robotic whisker using torque information; The International Journal of Robotics Research 29 (2010), pp. 1233–1245.

[29] C. Tuna, J. H. Solomon, D. L. Jones, and M. J. Z. Hartmann, Object shape recognition with artificial whiskers using tomographic reconstruction. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2012), pp. 2537-2540.

[30] N. Ueno, M. M. Svinin, and M. Kaneko, Dynamic contact sensing by flexible beam. IEEE/ASME Trans. Mechatronics 3(4) (1998), pp. 254-264.

[31] D. Voges, K. Carl, G. Klauer, R. Uhlig, C. Behn, C. Schilling, and H. Witte, Structural characterisation of the whisker system of the rat. IEEE Sensors Journal 12(2) (2012), pp. 332–339.

[32] W. Weaver, S. P. Timoshenko, and D. H. Young, Vibration Problems in Engineering. John Wiley & Sons Inc., Chichester (1990).

[33] L. E. Wineski, Facial morphology and vibrissal movement in the golden hamster. Journal of Morphology 183 (1985), pp. 199–217.

[34] H. Yokoi, M. Fend, and R. Pfeifer, Development of a Whisker Sensor System and Simulation of Active Whisking for Agent Navigation. In: Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and System (2004), pp. 607–612.

Towards Semantic Facility Data Management

Ilkka Niskanen, Anu Purhonen, Jarkko Kuusijärvi
Digital Service Research
VTT Technical Research Centre of Finland
Oulu, Finland
{Ilkka.Niskanen, Anu.Purhonen, Jarkko.Kuusijarvi}@vtt.fi

Esa Halmetoja,
Senate Properties
Oulu, Finland
Esa.Halmetoja@senaatti.fi

***Abstract*—Nowadays, facility management is realized with different information systems, which provide a comprehensive view for the management. Building Information Modelling (BIM) encompasses a computer model of a facility, which is utilized throughout the life-cycle of the building. To enable a more holistic view on facilities' conditions (e.g., energy efficiency, indoor environment, maintenance and repair) we present an approach which enhances the BIM model with semantic indoor measurement data. The enhanced semantic information provides more contextual information about the building; history of conditions, current conditions, and even predictions about the future conditions. In addition, the system ties facility users into the process of facility management by allowing them to view the current indoor conditions and give feedback about the conditions of the building. The resulting semantic facility data management approach was tested in an experiment in which the system was applied in a school building environment.**

Keywords—facility management, semantic technologies, BIM, user-awareness, indoor conditions, sensor measusrements

I. INTRODUCTION

The field of facility management refers to the coordination and maintenance of physical spaces and infrastructures such as office buildings, schools, hotels and government institutions. Efficient facility management requires understanding and engaging different stakeholders including building users, owners and operators. Additionally, although good facility management is traditionally measured by a reduced operating cost, more attention has been given to the impact of the overall qualitative aspects of the work environment on users' perceived satisfaction and ability to work [1].

The requirements of facility management have increased tremendously during the recent years. Especially the growing role of computerized support systems has led to more complicated facility management operations. For example, Building Information Modelling (BIM) has attained widespread attention [2]. BIM represents the process of development and use of a computer generated model to simulate the planning, design, construction and operation of a facility [3]. BIM models are computer generated data-rich and object-oriented representations of facilities from which views and data appropriate to various users' needs can be extracted and analysed to generate information that can be used to make decisions [4]. For facility management's perspective, the BIM models are useful especially for renovations, space planning and maintenance operations [3].

The recent technological advances in pervasive computing and wireless sensors have enabled also new types of facility services. The examples of applications are span from security and surveillance to monitoring of consumption of facility resources (e.g., measuring, logging and comparing water and electricity consumptions). Novel types of building performance measurement methods such as sensor network systems allow extensive heterogeneous information generated within facilities providing valuable information about the current state of a building. While extensive sensor data is collected from different environments there are still significant challenges in converting such data into useful information needed by different facility stakeholders [5].

The utilization of semantic technologies facilitates the management and interpretation of data collected from facilities. For example, the use of resource describing metadata enables more intelligent machine-to-machine interactions, such as reasoning, deduction and semantic searches [6][21]. Moreover, the abilities to merge heterogeneous data and derive high-level context information from low-level measurement data expand the scope of use of semantic technologies in the domain of facility management. While the quantity of data represented with semantic techniques has increased enormously, powerful database techniques for storing, managing and querying semantic data have been developed both by research community and industry [7][8][9].

Although there have been several approaches to utilize semantic technologies in the field of facility management [10][11][12], the potential of semantics is still yet to be fully realized. For example, the benefits deriving from the integration of static BIM data to dynamic facility monitoring data are not extensively exploited or understood. Additionally, more information about field tests and experiments in which these emerging technologies are applied in practical real-world settings taking into account the users' satisfaction perspective is sorely needed.

In this paper, a novel approach for semantic facility data management is introduced. The approach integrates and interprets facility information collected from heterogeneous sources and represents it for different stakeholders, including facility users, maintenance workers and owners. Furthermore, the approach allows facility users to give feedback about the conditions of a building.

The semantic data management approach was tested in an experiment in which it was applied in the Tervaväylä School. Tervaväylä is a state-funded special school and centre for development in special needs education and it is located in Oulu, Finland.

Besides the semantic data management approach, the test environment included a building automation system and a wireless sensor network that provided sensor-based measurement data, a server machine that hosted a semantic database and a tablet computer that held a Graphical User Interface (GUI) that allowed the users to examine the visualized facility information and interact with the approach.

The results of the experiment show that with the semantic facility data management approach it is possible to effectively merge and interpret heterogeneous facility data and produce interactive visualizations for different stakeholders. Furthermore, field tests conducted as a part of the experiment indicate that the possibility of examining sensor-based condition (temperature, energy consumption, etc.) information is perceived as a useful feature by the facility users. Moreover, the user interface that allows users to navigate through the school building and give feedback on the conditions of different rooms were found to increase user satisfaction. Additionally, the facility maintenance workers perceived the system as a valuable information resource that offers potential to support their daily activities, work processes and interaction towards the facility users. The suggestions for improvements that were derived from the experiment include fine-tuning the GUI and the visualizations provided by the approach.

The rest of paper is organized as follows. Section II gives a description of the test environment. In Section III the results of the field tests are discussed in more detail. Section IV concludes the paper.

II. TEST ENVIRONMENT

In Fig. 1, the different components of the test environment are presented.

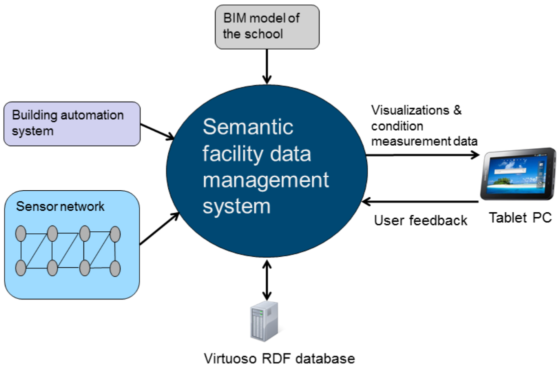


Fig. 1. The test environment used in the experiment

A. Building automation system

The building automation system in Tervaväylä School was accessed via RAUInfo [13], which is a service designed for the owners and maintenance workers of properties and which provides continuous monitoring data accessible via web-service interface. RAUInfo offers comprehensive monitoring data on, for example, heating, cooling and water and energy consumption of Tervaväylä School.

B. Wireless sensor network

The facility data acquired from the building automation system is augmented with additional measurement data provided by sensors mounted to selected rooms in Tervaväylä School. The sensors were installed to spaces that were uncovered by the building automation system but are actively used by the facility users. The additional sensors provide the following measurement values: temperature, illuminance, carbon dioxide level, moisture, and humidity.

C. BIM model of the school

The BIM model of the Tervaväylä School contains information about physical and functional characteristics of the facility. The BIM model represents the design of the building including spaces, objects and other building components. The integration of the BIM model into the overall system architecture provides several benefits. For example, by using the BIM model the different data providing sensors can be located and discovered more easily.

D. Virtuoso RDF database

The semantically described facility data is stored to Virtuoso [9], which is a database management system for RDF [14] data. Virtuoso offers numerous data access and storage mechanisms and interfaces. Virtuoso has been widely used platform and is continually developed further and is thus mature enough solution as the RDF database for the facility data. In addition, Virtuoso supports the storage and querying of very large datasets, which is essential in this context, since building automation and additional sensors can provide a large amount of information.

E. Semantic facility data management approach

The architecture of the semantic facility data management approach contains three main layers: a data collection and storing layer, a data processing layer, and a data representation layer. The data collection and storing layer is responsible for acquiring, semantically annotating and finally storing the facility-related data into the semantic database. The data processing layer enables interpreting semantically described facility data into more meaningful context information. For example, it realizes SPARQL [15] querying functionalities, manages different user profiles and provides necessary information for visualization views. The semantic data processing capabilities provided by the data collection and storing layer as well as the data processing layer are enhanced by the extensive utilization of ontologies.

Ontologies are commonly used to formally represent a set of concepts within a domain and the relationships between pairs of concepts. Ontologies support modelling a domain and performing reasoning about different entities. Ontologies also specify a shared vocabulary and taxonomy which represent a domain including its concepts and their properties and relations [17]. In semantic facility data management approach, ontologies are utilized for formally representing domain specific concepts and their relationships, and metadata that enables the system to better understand, and reason about the structure and purpose of the data. Moreover, ontologies enable

 ISBN: 978-1-61208-352-0

the integration of various data sources by resolving semantic heterogeneity between them.

To support the functionality of the framework, three novel ontology definitions were constructed: Building ontology, Sensor ontology and Feedback ontology. The ontologies are described in more detail in the following sub-chapters.

a) Building ontology

To enable the semantic modelling of data contained within BIM models a new ontology was developed. The resulting ontology for semantically storing BIM data is sketched in Fig. 2.

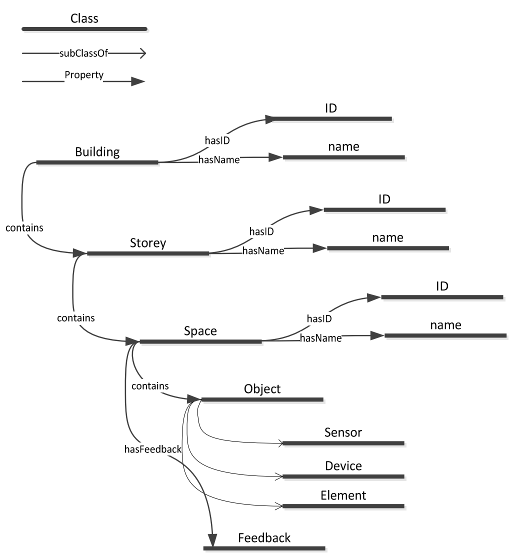


Fig. 2. Building ontology

As presented in Fig. 2, the Building ontology contains four main classes – Storey, Space, Object and Feedback. The next level of the ontology contains subclasses that represent different types of objects, for example, devices. Moreover, each sub class holds its own object type specific properties that provide more specific characteristics about the entities they represent. The Sensor and Feedback classes represent linkages to the other ontologies which are described in more detail later in this section.

Currently, there exist some [18][19][20] approaches that define their own ontologies for semantically describing BIM models. However, for this study it was decided to design a new BIM ontology that adopts some elements from the existing approaches but is especially adapted and optimised for visualisation and monitoring purposes. This more lightweight and flexible ontology is unencumbered by the burden of semantically describing all the concepts and content contained by BIM models. On the other hand, the defined ontology structure offers enough expressiveness for providing comprehensive visualizations and performing sophisticated diagnosis and analysis operations. Additionally, the BIM ontology is general enough to be easily expandable for future needs.

b) Sensor ontology

The system defines a sensor ontology to enable semantic modelling of sensor-based measurement data. Additionally, the sensor ontology facilitates the extraction of high-level context information from various streams of continuous sensor data. In Fig. 3, the designed sensor ontology is presented in more detail.

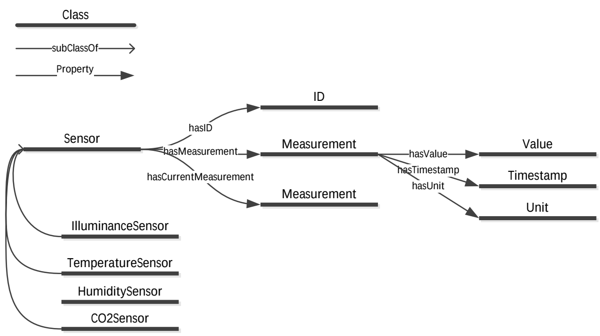


Fig. 3. Sensor ontology

The ontology contains the different types of sensors and their measurements. Every measurement has a timestamp, unit, and value, which are used, e.g., in visualizing the measurements. Each sensor is attached to a specific location in a room in the Building ontology. The location of the sensor can be used in analysing the indoor environment of the target building and to make reasoning about possible events, e.g., heating failure, that would need maintenance

c) Feedback ontology

The feedback provided by facility users is modelled and stored using an ontology description. In Fig. 4 the feedback ontology is presented in more detail.

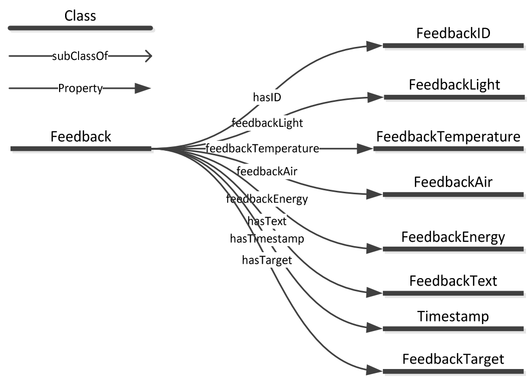


Fig. 4. Feedback ontology

The feedback ontology includes values for the specified feedback attributes: lighting, temperature, air, energy, and free text. The free text field can be used to give plain text feedback about the space. The other fields are numerical values ranging from 0 to 100. For example, in the temperature field a value of 0 corresponds to very cold and a value of 100 corresponds to very hot. The end-user uses a slider with visual cues and a text describing the situation. The feedback is tied to a specific location target (e.g., room) in the building. The feedbacks are given a timestamp so that they can be easily compared to the measured indoor conditions from the building automation and additional sensors.

d) Graphical User Interface

The role of the data representation layer is to implement the GUI, which is responsible for creating visualization views and managing interaction between the end-users and the system. The GUI is implemented with HTML5 [16] utilizing graphical libraries optimized for mobile devices. The application is a full web-based application run with a web browser; no native programming language was used. Thus, the application can be used in multiple platforms and devices ranging from computers to tablets and mobile phones. In this experiment, a Samsung Galaxy Tab 10.1 and a laptop were used.

The objective of the GUI is to enhance user awareness by providing means to explore building-related data through visualizations that represent different aspects of the building. Different spaces of the building are represented from an isometric perspective, which provides an overview of the contextual environment and facilitates the discovery of spatial relationships between objects. Moreover, the isometric visualization contains a summary of the measurement information that is provided by the different sensors that are monitoring the conditions of spaces in real-time. The Isometric visualization of a space is shown in Fig. 5.

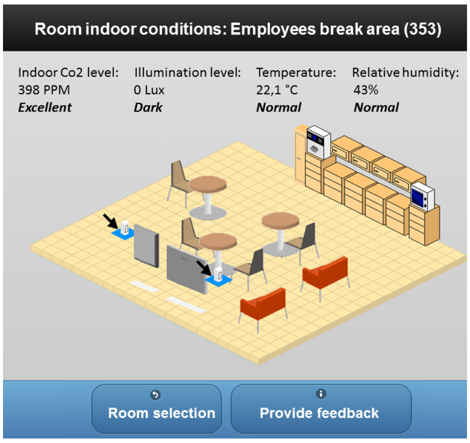


Fig. 5. A visualization of a space

From the isometric visualization the user is able to access a feedback section in which the user can give either general feedback or feedback concerning the existing indoor conditions of a certain space. The approach enables giving either verbal or scaled feedback. Scaled feedback is given by using special sliding clutches in estimating the current status (e.g., from too warm to cold) of four parameters that are temperature, air quality, lighting and energy consumption.

The facility maintenance workers are able to see indoor environment conditions (e.g., temperature, humidity, etc.) of the selected room in line chart visualizations over a selected period of time, e.g., a day or a week (see Fig. 6). Additionally, the facility maintenance workers can view the received feedbacks from the facility users to specific locations in the building.

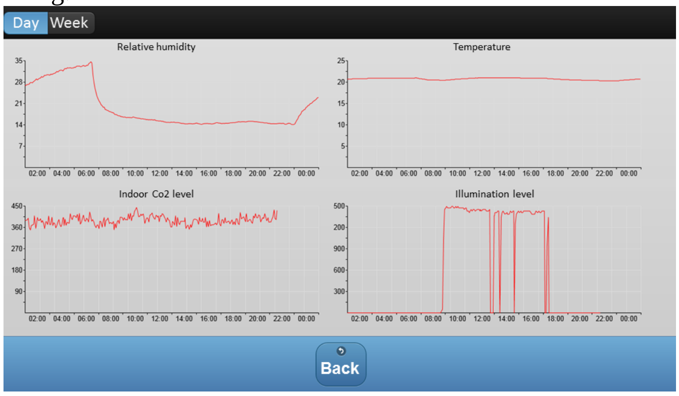


Fig. 6. A historical condition data representation

III. Field Tests

The field tests were conducted in order to achieve the following objectives:

- To validate the functionality of the semantic facility data management approach in real-world settings.
- To examine the usage rate of the semantic facility data management approach among the users.
- To study the effects of the semantic facility data management approach on facility user satisfaction.
- To provide information about how the facility users perceive the existing conditions of the school.

A. Experiment execution

During the field test periods the tablet running the GUI of the system was located in the school employees' break room, where it was available for the personnel to use. The number of people involved in the experiment was approximately 55. In total, three separate test periods were performed and between each period the semantic facility data management approach was improved according to the received user feedback. The times of the test periods are shown below.

Field test 1: 4/12/2012 – 4/1/2013

Field test 2: 1/2/2013 – 15/2/2013

Field test 3: 26/3/2013 – 12/4/2013

B. Experiment results

After the field tests the usage metrics were analysed. Furthermore, an additional questionnaire was prepared in order to measure the perceived ease-of-use and perceived usefulness of the approach. The questionnaire used a five level grading system, where five is the best and one the worst grade, three being the average. Users were able to answer the questionnaire anonymously.

 ISBN: 978-1-61208-352-0

1) Results of usage metrics

The usage metrics indicate that the interest towards the approach was at its peak during the first field test period, in which the different features were used the most frequently. During the second and the third field test periods the users were probably more familiar with the approach and hence used it only when they were interested in the conditions of a certain room or wanted to give feedback about a specific deficiency, for example. A condensed summary of the data obtained from the usage metrics are shown in Table I.

TABLE I. SUMMARIZED USAGE METRICS DATA

	Field test 1	Field test 2	Field test 3
Main page opened	164	12	11
Info page opened	102	40	21
Feedback sent	18 (of which 7 written)	2 (0)	5 (3)

1) Results of the numerical feedback

According to the numerical feedback received the school employees were mainly satisfied with the indoor environment. However, temperature conditions in different spaces received some negative feedback. When comparing the negative feedback to other facility data managed with the approach it was discovered that the outside weather had an effect on how people perceived the indoor temperature. More precisely, during a cold winter day the indoor temperatures were usually perceived too low whereas a sunny day had an opposite impact. Additionally, when analysing the visualizations of different spaces it was discovered that the negative temperature feedbacks were focused on rooms that contained large windows, which apparently strengthen the effect of warm or cold outside temperatures.

2) Results of the questionnaire

To summarize the questionnaire results, the visualizations offered by the semantic facility data management approach were considered as an important and useful information source by the facility users. However, the usability of the approach was found to require improvements. Moreover, it was perceived as difficult to give written feedback with the tablet. In addition, some data representation techniques used by the visualizations were regarded as difficult to understand. The background knowledge of the end-users on using tablets probably has an effect on the overall satisfaction on the user interface.

The possibility to give feedback anonymously was considered a very positive feature. In more detail, the users felt more convenient to give feedback with a tablet than, e.g., a bigger info screen (if the user wanted to give textual feedback, he/she could do so privately with the tablet). However, according to the questionnaire results a small portion of the employees were not interested in learning to use the tablet PC or the GUI of the approach, which hindered their participation.

3) Feedback from the facility maintenance workers

Besides the field test periods, the approach was introduced to the facility maintenance workers of the Tervaväylä School. The facility maintenance workers were familiarized with the approach and they were given an opportunity to test it. Afterwards, the facility maintenance workers answered a questionnaire that measured the perceived ease-of-use and perceived usefulness.

According to the questionnaire results, the tablet was considered as a useful tool and the user interface of the approach was perceived as clear and easy to use. The possibility to receive feedback directly from the users of the building was considered as an interesting feature. However, the facility maintenance workers were a bit concerned whether they have enough resources to react on every comment made through the system. A suggestion made by the facility maintenance workers was to use some kind of filter to extract the most important notices of defects or service requests from the received feedbacks.

In general, the facility maintenance workers appreciated the idea of having a single interface that is used for observing the information of the building. Currently, the information that they need is scattered in three different systems. Also graphical representations of sensor-based measurement data received positive comments. However, according to the facility maintenance workers the approach should offer more flexibility in setting the time range for observing different sensor data measurements. The final conclusion that emerged from the questionnaire was that more advanced means to give additional information through the approach should be provided. For example, it would be useful to inform the facility users about the water or heating system outages or the testing of fire alarms.

IV. CONCLUSIONS AND FUTURE WORK

In this work, a novel approach for facility data management was introduced. The approach utilized semantic technologies to integrate heterogeneous facility data and interactive visualizations to improve user awareness. The approach also offered efficient and easy-to-use mechanisms for providing feedback. Moreover, the approach aimed at aiding the work of facility maintenance workers by offering a unified interface to examine building data and to interact with the facility users.

Results of an experiment, in which the approach was used in real-world settings, were also presented in the paper. The results indicate that by providing real-time information about indoor environment in a meaningful form and by offering convenient ways to give feedback, the customer experience for the facility users can be improved. Furthermore, with the approach the facility maintenance workers are able to better adjust the conditions according to the needs of the users and be aware on users' perceived satisfaction and ability to work. Moreover, semantic techniques were found to be adequate in terms of performance and scalability in real world facility data management activities. Finally, it was discovered that the existing indoor conditions in Tervaväylä School are in a satisfactory level.

The future work includes improving the deficiencies found during the experiment as well as further developing the approach. The possible developing activities include, for example, integrating the approach with existing energy management systems, providing enhanced visualizations that facilitate the visibility of maintenance services to the facility users and utilizing the collected information in the proactive prevention of problems. Furthermore, in order to improve the abilities of the approach to support the tasks of different stakeholders (e.g., maintenance workers, facility owners) a more throughout analysis of their needs and requirements should be conducted.

REFERENCES

[1] D. Leifer, “Evaluating user satisfaction: case studies in Australasia,” Facilities, vol. 16 no. 5/6, pp. 138-142, 1998.

[2] U. Isikdag, J. Underwood, and M. Kuruoglu, “Building Information Modelling,” in A. Akintoye, J.S. Goulding, and G. Zawdie, (Eds.), Construction Innovation and Process Improvement, Wiley Blackwell, UK, 2012.

[3] A. Salman, M. Hein, and B. Sketo, "Building information modeling (BIM): Benefits, risks and challenges," Proc. 44th ASC National Conference, 2008.

[4] Associated General Contractors Guide, “AGC Contractors’ Guide to BIM”, 2006. (Available online at: http://www.tpm.com/wp-content/uploads/2013/02/AGC_Guide_to_BIM.pdf) [retrieved: May 2014]

[5] S. D. Glaser and A. Tolman, "Sense of sensing: From data to informed decisions for the built environment," Journal of infrastructure systems vol. 14, no. 1, pp. 4-14, 2008

[6] L. Baolin and B. Hu, "HPRD: a high performance RDF database." In Network and Parallel Computing, pp. 364-374, Springer Berlin Heidelberg, 2007.

[7] OpenRDF, http://www.openrdf.org/index.jsp [retrieved: May 2014]

[8] StarDog RDF database, http://stardog.com/ [retrieved: May 2014]

[9] Virtuoso RDF data store, http://virtuoso.openlinksw.com/rdf-quad-store/ [retrieved: May 2014]

[10] Q. Z. Yang and Y. Zhang, "Semantic interoperability in building design: Methods and tools," Computer-Aided Design vol. 38 no. 10 pp. 1099-1112, 2006.

[11] R. Vanlande, C. Nicolle, and C. Cruz, "IFC and building lifecycle management," Automation in Construction, vol. 18, no. 1 pp. 70-78, 2008.

[12] H. Schevers, J. Mitchell, P. Akhurst, D. Marchant, S. Bull, K. McDonald, R. Drogemuller, and C. Linning, "Towards digital facility modelling for Sydney opera house using IFC and semantic web technology," ITcon vol. 12, pp. 347-362, 2007.

[13] RAUInfo, http://www.rauinfo.fi/Intro.do [retrieved: May 2014]

[14] Resource Description Framework (RDF), http://www.w3.org/RDF/ [retrieved: May 2014]

[15] SPARQL specification, http://www.w3.org/TR/sparql11-query/ [retrieved: May 2014]

[16] M. Pilgrim, HTML5: Up and running, O'Reilly, 2010.

[17] T. Gruber, Toward “Principles for the Design of Ontologies Used for Knowledge Sharing,” International Journal Human-Computer Studies Vol. 43, Issues 5-6, Novemer 1995, pp. 907-928.

[18] C. Lima, T. El-Diraby, and J. Stephens, "Ontology-based optimization of knowledge management in e-construction," Journal of IT in Construction, vol. 10, 2005, pp. 305-327.

[19] P. Pauwels, R. De Meyer, and J. Van Campenhout, "Interoperability for the design and construction industry through semantic web technology," Semantic Multimedia. Springer Berlin Heidelberg, 2011, pp. 143-158.

[20] P. Pauwels, D. Van Deursen, R. Verstraeten, J. De Roo, R. De Meyer, R. Van de Walle, and J. Van Campenhout, "A semantic rule checking environment for building performance checking," Automation in Construction, vol 20, 2011, pp. 506-518.

[21] R, Albertoni and M. De Martino, "Semantic similarity of ontology instances tailored on the application context," On the Move to Meaningful Internet Systems, CoopIS, DOA, GADA, and ODBASE, Springer Berlin Heidelberg, 2006. pp. 1020-1038.

Complexity of Rule Sets Induced from Incomplete Data with Lost Values and Attribute-Concept Values

Patrick G. Clark
Department of Electrical Eng. and Computer Sci.
University of Kansas
Lawrence, KS, USA
e-mail: patrick.g.clark@gmail.com

Jerzy W. Grzymala-Busse
Department of Electrical Eng. and Computer Sci.
University of Kansas
Lawrence, KS, USA
Institute of Computer Science
Polish Academy of Sciences
Warsaw, Poland
e-mail: jerzy@ku.edu

***Abstract*—This paper presents novel research on complexity of rule sets induced from incomplete data sets with two interpretations of missing attribute values: lost values and attribute-concept values. Experiments were conducted on 176 data sets, using three kinds of probabilistic approximations (lower, middle and upper) and the Modified Learning from Examples Module, version 2 (MLEM2) rule induction system. In our experiments, the size of the rule set was always smaller for attribute-concept values than for lost values (5% significance level). The total number of conditions was smaller for attribute-concept values than for lost values for 17 combinations of the type of data set and approximation, out of 24 combinations total. In remaining 7 cases, the difference in performance was statistically insignificant. Thus, we may claim that attribute-concept values are better than lost values in terms of rule complexity.**

Keywords–Data mining; rough set theory; probabilistic approximations; MLEM2 rule induction algorithm; lost values; attribute-concept values.

I. Introduction

Standard lower and upper approximations are fundamental concepts of rough set theory. A probabilistic approximation, associated with a probability α, is a generalization of the standard approximation. For α = 1, the probability approximation is reduced to the lower approximation; for very small α, it is reduced to the upper approximation. Research on theoretical properties of probabilistic approximations started from [1] and then was continued in many papers, see, e.g., [1]–[6].

Incomplete data sets may be analyzed using global approximations such as singleton, subset and concept [7][8]. Probabilistic approximations, for incomplete data sets and based on an arbitrary binary relation, were introduced in [9], while first experimental results using probabilistic approximations were published in [10].

In this paper, incomplete data sets are characterized by missing attribute values. We will use two interpretations of a missing attribute value: lost values and attribute-concept values.

For our experiments we used 176 incomplete data sets, with two types of missing attribute values: lost values and attribute-concept values. Additionally, in our experiments we used three types of approximations: lower, upper, and additionally the most typical probabilistic approximation, for α = 0.5, called a middle approximation.

From our previous research it follows that the correctness of the rule sets, evaluated by ten-fold cross validated error rate, do not differ significantly with different combinations of missing attribute and approximation type.

In our experiments, the size of rule set was always smaller for attribute-concept values than for lost values. The total number of conditions in rule sets was smaller for attribute-concept values for 17 combinations of the type of data set and approximation (out of 24 combinations total). In remaining seven combinations, the total number of conditions in rule sets did not differ significantly. Thus, we may claim that attribute-concept values are better than lost values in terms of rule complexity.

Our secondary objective was to check which approximation (lower, middle or upper) is the best from the point of view of rule complexity.

The smallest size of rule sets was accomplished, in five (out of 24 combinations) for lower approximations and in two combinations for upper approximations. The total number of conditions in rule sets was achieved, again, for lower approximations in five combinations and for upper approximations in other two combinations. For remaining 17 combinations the difference between all three approximations was insignificant.

This paper starts with a discussion on incomplete data in Section II where we define approximations, attribute-value blocks and characteristic sets. In Section III, we present probabilistic approximations for incomplete data. Section IV contains the details of our experiments. Finally, conclusions are presented in Section V.

II. Incomplete Data

We assume that the input data sets are presented in the form of a decision table. An example of a decision table is shown in Table I. Rows of the decision table represent cases, while columns are labeled by variables. The set of all cases will be denoted by U. In Table I, $U = \{1, 2, 3, 4, 5, 6, 7, 8\}$. Independent variables are called attributes and a dependent variable is called a decision and is denoted by d. The set of all attributes will be denoted by A. In Table I, $A = \{$*Education*, *Skills*, *Experience*$\}$. The value for a case x and an attribute a will be denoted by $a(x)$.

 ISBN: 978-1-61208-352-0

TABLE I. A DECISION TABLE

	Attributes			Decision
Case	Education	Skills	Experience	Productivity
1	higher	high	–	high
2	?	high	low	high
3	secondary	–	high	high
4	higher	?	high	high
5	elementary	high	low	low
6	secondary	–	high	low
7	–	low	high	low
8	elementary	?	–	low

In this paper, we distinguish between two interpretations of missing attribute values: lost values and attribute-concept values. Lost values, denoted by "?", mean that the original attribute value is no longer accessible and that during rule induction we will only use existing attribute values [11][12]. Attribute-concept values, denoted by "–", mean that the original attribute value is unknown; however, because we know the concept to which a case belongs, we know all possible attribute values. Table I presents an incomplete data set affected by both lost values and attribute-concept values.

One of the most important ideas of rough set theory [13] is an indiscernibility relation, defined for complete data sets. Let B be a nonempty subset of A. The indiscernibility relation *R*(B) is a relation on U defined for $x, y \in U$ as defined in equation 1.

$$(x, y) \in R(B) \text{ if and only if } \forall a \in B \ (a(x) = a(y)) \quad (1)$$

The indiscernibility relation $R(B)$ is an equivalence relation. Equivalence classes of $R(B)$ are called *elementary sets* of B and are denoted by $[x]_B$. A subset of U is called *B-definable* if it is a union of elementary sets of B.

The set X of all cases defined by the same value of the decision d is called a *concept*. For example, a concept associated with the value *low* of the decision *Productivity* is the set $\{1, 2, 3, 4\}$. The largest B-definable set contained in X is called the *B-lower approximation* of X, denoted by $\underline{appr}_B(X)$, and defined in equation 2.

$$\cup\{[x]_B \mid [x]_B \subseteq X\} \quad (2)$$

The smallest B-definable set containing X, denoted by $\overline{appr}_B(X)$ is called the *B-upper approximation* of X, and is defined in equation 3.

$$\cup\{[x]_B \mid [x]_B \cap X \neq \emptyset\} \quad (3)$$

For a variable a and its value v, (a, v) is called a variable-value pair. A *block* of (a, v), denoted by $[(a, v)]$, is the set $\{x \in U \mid a(x) = v\}$ [14]. For incomplete decision tables the definition of a block of an attribute-value pair is modified in the following way.

- If for an attribute a there exists a case x such that $a(x) = ?$, i.e., the corresponding value is lost, then the case x should not be included in any blocks $[(a, v)]$ for all values v of attribute a,
- If for an attribute a there exists a case x such that the corresponding value is an attribute-concept value, i.e., $a(x) = -$, then the corresponding case x should be included in blocks $[(a, v)]$ for all specified values $v \in V(x, a)$ of attribute a, and is defined by equation 4.

$$V(x, a) = \{a(y) \mid a(y) \ is\ specified,\ y \in U,\ d(y) = d(x)\} \quad (4)$$

For the data set from Table I, the attribute-concept values are defined as: $V(1, Experience) = \{low, high\}$, $V(3, Skills) = \{high\}$, $V(6, Skills) = \{low, high\}$, $V(7, Education) = \{elementary, secondary\}$ and $V(8, Experience) = \{low, high\}$.

For the data set from Table I the blocks of attribute-value pairs are: [(Education, elementary)] = $\{5, 7, 8\}$, [(Education, secondary)] = $\{3, 6, 7\}$, [(Education, higher)] = $\{1, 4\}$, [(Skills, low)] = $\{6, 7\}$, [(Skills, high)] = $\{1, 2, 3, 5, 6\}$, [(Experience, low)] = $\{1, 2, 5, 8\}$, and [(Experience, high)] = $\{1, 3, 4, 6, 7, 8\}$.

For a case $x \in U$ and $B \subseteq A$, the *characteristic set* $K_B(x)$ is defined as the intersection of the sets $K(x, a)$, for all $a \in B$, where the set $K(x, a)$ is defined in the following way:

- If $a(x)$ is specified, then $K(x, a)$ is the block $[(a, a(x))]$ of attribute a and its value $a(x)$,
- If $a(x) =?$ then the set $K(x, a) = U$, where U is the set of all cases,
- If $a(x) = -$, then the corresponding set $K(x, a)$ is equal to the union of all blocks of attribute-value pairs (a, v), where $v \in V(x, a)$ if $V(x, a)$ is nonempty. If $V(x, a)$ is empty, $K(x, a) = U$.

For Table I and $B = A$, $K_A(1) = \{1\}$, $K_A(2) = \{1, 2, 5\}$, $K_A(3) = \{3, 6\}$, $K_A(4) = \{1, 4\}$, $K_A(5) = \{5\}$, $K_A(6) = \{3, 6, 7\}$, $K_A(7) = \{6, 7\}$, and $K_A(8) = \{5, 7, 8\}$.

Note that for incomplete data there are a few possible ways to define approximations [7], we used *concept* approximations [9] since our previous experiments indicated that such approximations are most efficient [9]. A B-*concept lower approximation* of the concept X is defined in equation 5.

$$\underline{B}X = \cup\{K_B(x) \mid x \in X, K_B(x) \subseteq X\} \quad (5)$$

The *B-concept upper approximation* of the concept X is defined by the equation 6.

$$\overline{B}X = \cup\{K_B(x) \mid x \in X, K_B(x) \cap X \neq \emptyset\} = \cup\{K_B(x) \mid x \in X\} \quad (6)$$

For Table I, A-concept lower and A-concept upper approximations of the concept $\{1, 2, 3, 4\}$ are $\underline{A}\{1, 2, 3, 4\} = \{1, 4\}$ and $\overline{A}\{1, 2, 3, 4\} = \{1, 2, 3, 4, 5, 6\}$, respectively.

III. PROBABILISTIC APPROXIMATIONS

For completely specified data sets a *probabilistic approximation* is defined by equation 7, where α is a parameter, $0 < \alpha \leq 1$, see [1][4][9][15]–[17]. Additionally, for simplicity, the elementary sets $[x]_A$ are denoted by $[x]$. For discussion on how this definition is related to the value precision asymmetric rough sets see [9][10].

 ISBN: 978-1-61208-352-0

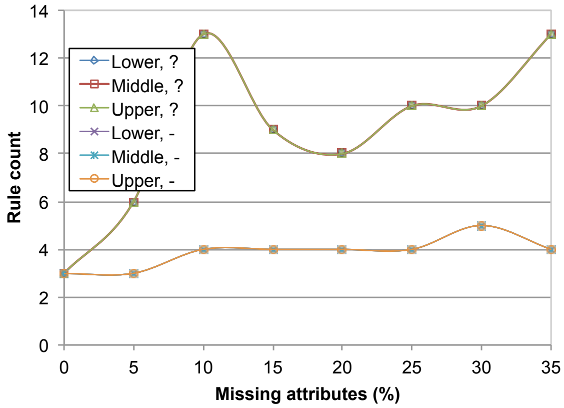


Figure 1. Size of the rule set for the *Bankruptcy* data set

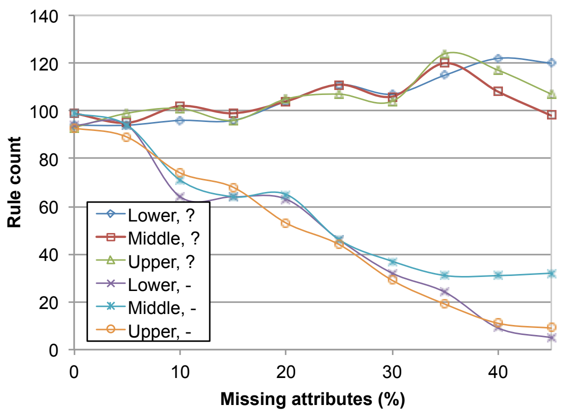


Figure 2. Size of the rule set for the *Breast cancer* data set

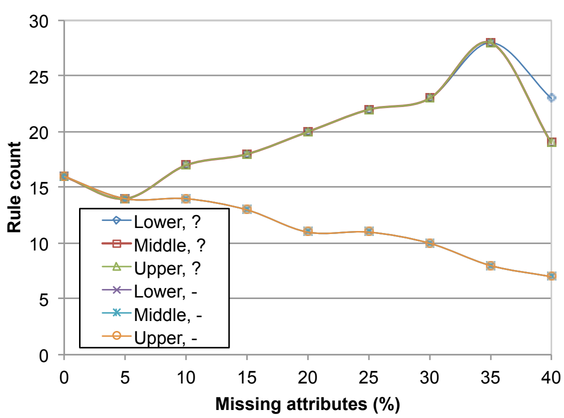


Figure 3. Size of the rule set for the *Echocardiogram* data set

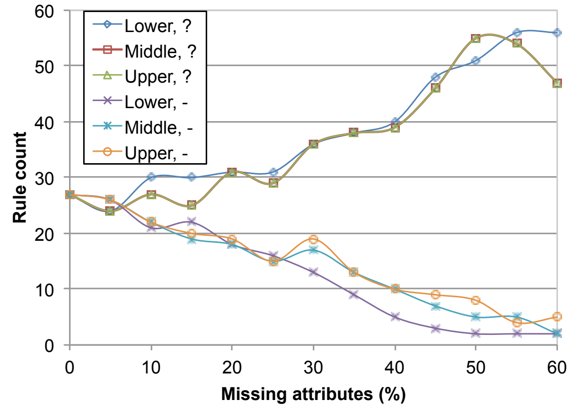


Figure 4. Size of the rule set for the *Hepatitis* data set

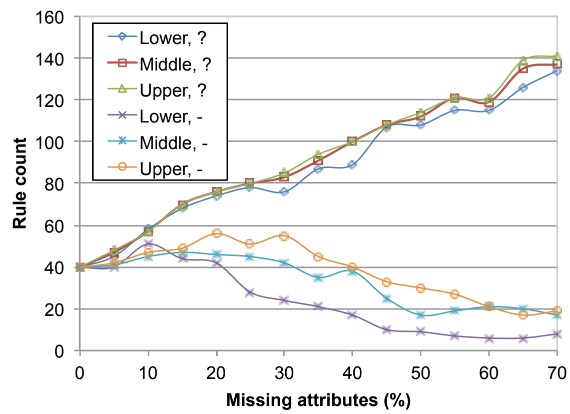


Figure 5. Size of the rule set for the *Image segmentation* data set

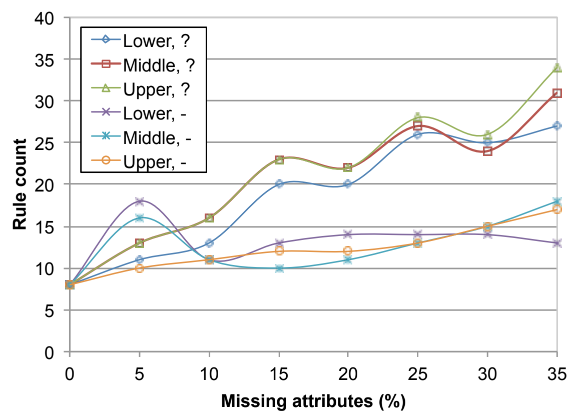


Figure 6. Size of the rule set for the *Iris* data set

$$appr_{\alpha}(X) = \cup\{[x] \mid x \in U, P(X \mid [x]) \geq \alpha\} \quad (7)$$

Note that if $\alpha = 1$, the probabilistic approximation becomes the standard lower approximation and if α is small, close to 0, in our experiments it was 0.001, the same definition describes the standard upper approximation.

For incomplete data sets, a *B-concept probabilistic approximation* is defined by equation 8 [9].

$$\cup\{K_B(x) \mid x \in X,\ Pr(X|K_B(x)) \geq \alpha\} \quad (8)$$

For simplicity, we will denote $K_A(x)$ by $K(x)$ and the *A-concept probabilistic approximation* will be called a *probabilistic approximation*.

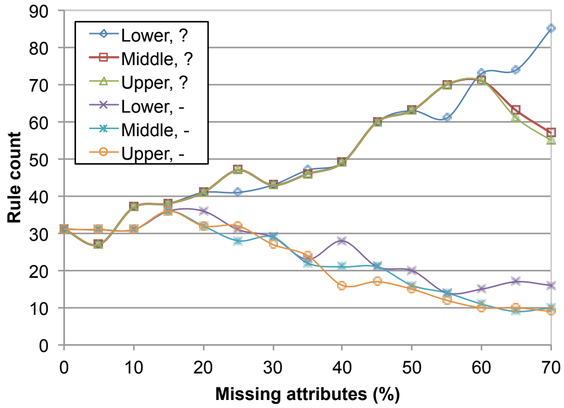


Figure 7. Size of the rule set for the *Lymphography* data set

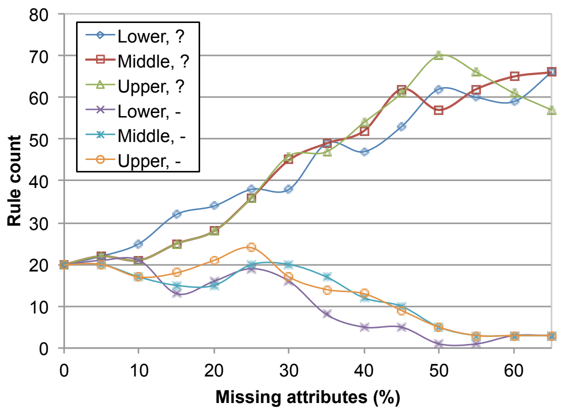


Figure 8. Size of the rule set for the *Wine recognition* data set

The special probabilistic approximations with the parameter $\alpha = 0.5$ will be called a *middle* approximation.

IV. EXPERIMENTS

Our experiments are based on eight data sets that are available on the University of California at Irvine *Machine Learning Repository*.

For every data set a set of templates was created. Templates were formed by replacing incrementally (with 5% increment) existing specified attribute values by *lost values*. Thus, we started each series of experiments with no *lost values*, then we added 5% of *lost values*, then we added additional 5% of *lost values*, etc., until at least one entire row of the data sets was full of *lost values*. Then three attempts were made to change configuration of new *lost values* and either a new data set with extra 5% of *lost values* was created or the process was terminated. Additionally, the same formed templates were edited for further experiments by replacing question marks, representing *lost values* by"−"s representing *attribute-concept values*.

For any data set there was some maximum for the percentage of missing attribute values. For example, for the *bankruptcy* data set, it was 35%. Hence, for the *bankruptcy* data set, we created seven data sets with lost values and seven data sets with attribute-concept values, for the total of 15 data sets (the additional data set was complete, with no missing attribute values). By the same token, for the *breast cancer*, *echocardiogram*, *hepatitis*, *image segmentation*, *iris*, *lymphography* and *wine recognition* data sets we created 19, 17, 25, 29, 15, 29, and 27 data sets. The total number of the data sets was 176.

Results of our experiments are presented in Figures 1–16.

We compared two interpretations of missing attribute values, lost values and attribute-concept values, assuming the same type of approximations. More explicitly, we compared the complexity of rule sets, first the size of rule sets, then the total number of conditions in the rule set, separately for lower approximations, then for middle approximations, and finally, for upper approximations, using the Wilcoxon matched-pairs signed rank test, with the 5% level of significance for two-tailed test.

For all eight types of data sets and all three types of approximations, the rule set size was always smaller for attribute-concept values than for lost values. For the total number of conditions in the rule sets results were more complicated. The total number of conditions in the rule sets was smaller for attribute-concept values than for lost values for 17 combinations of the type of data set and approximation, out of 24 possible combinations. For *echocardiogram* and *iris* data sets, for all three types of approximations and for the *lymphography* data set and lower approximations, the total number of conditions in rule sets for both interpretations of missing attribute values, did not differ significantly.

We compared all three types of approximations as well, assuming the same interpretation of missing attribute values, in terms of the size of rule sets and the total number of conditions in rule sets, using the Friedman Rank Sums test, again, with 5% of significance level.

The size of the rule set was smaller for lower approximations than for upper approximations for three combinations of the type of data set and type of missing attribute values (for the *hepatitis* data set and attribute-concept values and for the *image segmentation* data set and both lost values and attribute-concept values). The size of the rule set was smaller for lower approximations than for middle approximations in two combinations of the type of data set and type of missing attribute value (for the *image segmentation* data set and both lost values and attribute-concept values). Thus, for five combinations (out of 24) lower approximations were better than other approximations. On the other hand, the size of the rule set was smaller for upper approximations than for lower approximations for one combination (for the *lymphography* data set and attribute-concept values). Additionally, the size of the rule set was smaller for upper approximations than for middle approximations also for one combination (for the *breast cancer* data set and the attribute-concept values). Thus, for two combinations (out of 24) upper approximations were better than other approximations. For remaining 17 combinations the difference between all three approximations was insignificant.

The total number of conditions in rule sets was smaller for lower approximations than for upper approximations in four combinations of the type of data set and type of missing attribute value (for the *hepatitis* data set and attribute-concept

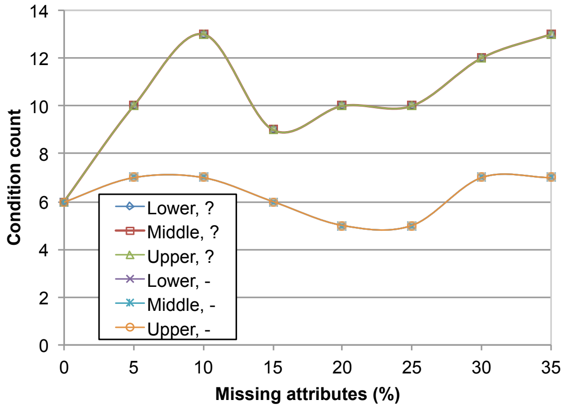


Figure 9. Number of conditions for the *Bankruptcy* data set

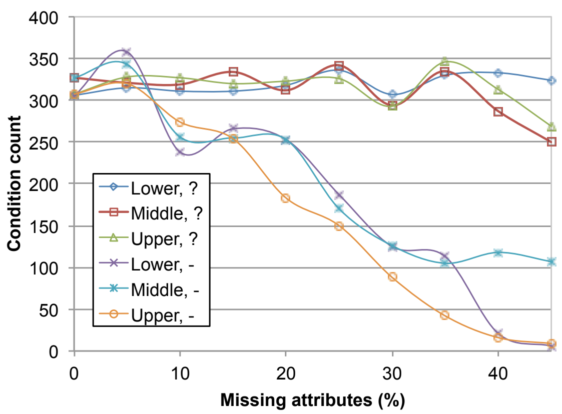


Figure 10. Number of conditions for the *Breast cancer* data set

values and for the *image segmentation* data set and both lost values and attribute-concept values and for the *iris* data set and lost values). The total number of conditions in rule sets was smaller for lower approximations than for middle approximations in one combination (for the *image segmentation* data set and lost values). Thus, for five combinations (out of 24) lower approximations were better than other approximations. The total number of conditions in rule sets was smaller for middle approximations than for lower approximations for one combination (for the *lymphography* data set and the attribute-concept values). Additionally, the total number of conditions in rules sets was smaller for upper approximations than for lower approximations also for one combination (for the *lymphography* data set and the attribute-concept values). Thus, for two combinations (out of 24) other approximations were better than lower approximations. For remaining 17 combinations the difference between all three approximations was insignificant.

In our experiments, we used the MLEM2 rule induction algorithm of the Learning from Examples using Rough Sets (LERS) data mining system [10][18][19].

V. Conclusions

As follows from our experiments, the size of rule set was always smaller for attribute-concept values than for lost values. The total number of conditions in rule sets was smaller for

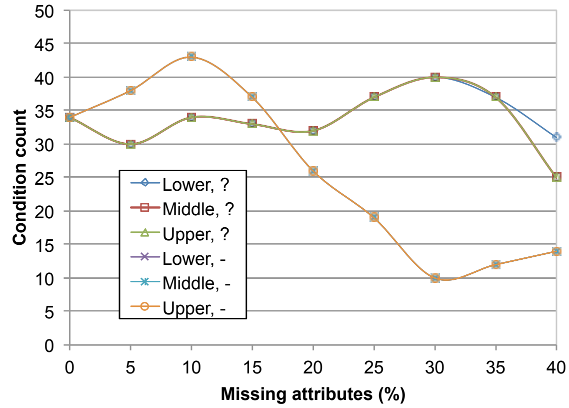


Figure 11. Number of conditions for the *Echocardiogram* data set

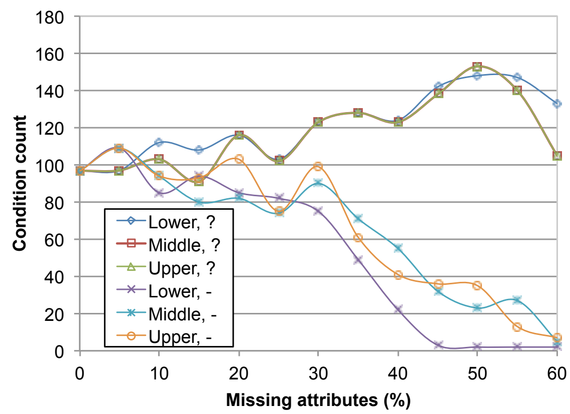


Figure 12. Number of conditions for the *Hepatitis* data set

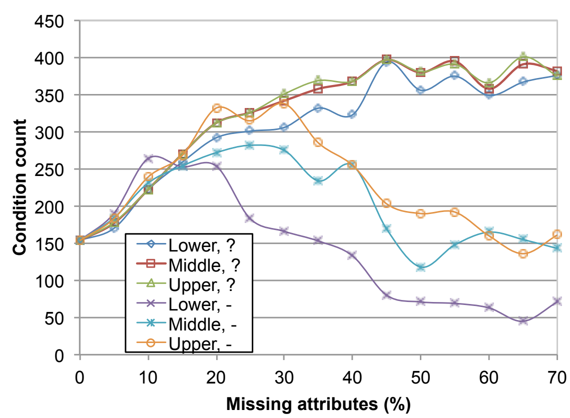


Figure 13. Number of conditions for the *Image segmentation* data set

attribute-concept values for 17 combinations of the type of data set and approximation (out of 24 combinations total). In remaining seven combinations, the total number of conditions in rule sets did not differ significantly. Thus, we may claim attribute-concept values are better than lost values in terms of rule complexity.

The smallest size of rule sets was accomplished, in five

 ISBN: 978-1-61208-352-0

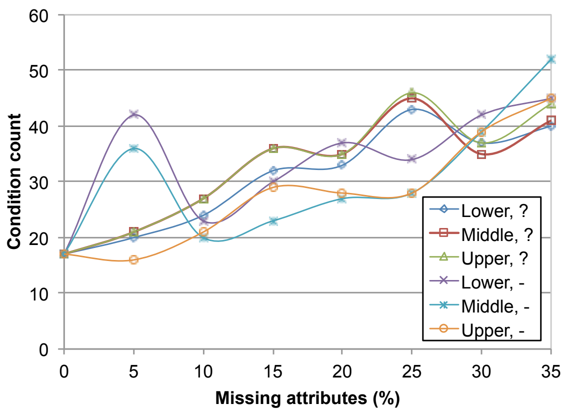


Figure 14. Number of conditions for the *Iris* data set

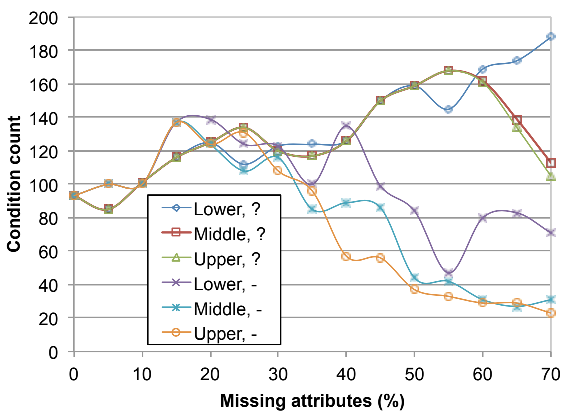


Figure 15. Number of conditions for the *Lymphography* data set

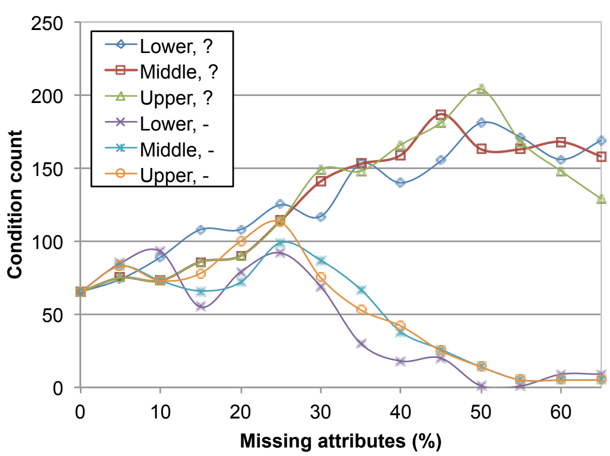


Figure 16. Number of conditions for the *Wine recognition* data set

(out of 24 combinations for lower approximations and in two combinations for upper approximations. The total number of conditions in rule sets was achieved, again, for lower approximations in five combinations and for upper or middle approximations in other two combinations. For remaining 17 combinations the difference between all three approximations was insignificant.

REFERENCES

[1] Z. Pawlak, S. K. M. Wong, and W. Ziarko, "Rough sets: probabilistic versus deterministic approach," International Journal of Man-Machine Studies, vol. 29, 1988, pp. 81–95.

[2] Z. Pawlak and A. Skowron, "Rough sets: Some extensions," Information Sciences, vol. 177, 2007, pp. 28–40.

[3] D. Ślęzak and W. Ziarko, "The investigation of the bayesian rough set model," International Journal of Approximate Reasoning, vol. 40, 2005, pp. 81–91.

[4] Y. Y. Yao, "Probabilistic rough set approximations," International Journal of Approximate Reasoning, vol. 49, 2008, pp. 255–271.

[5] Y. Y. Yao and S. K. M. Wong, "A decision theoretic framework for approximate concepts," International Journal of Man-Machine Studies, vol. 37, 1992, pp. 793–809.

[6] W. Ziarko, "Probabilistic approach to rough sets," International Journal of Approximate Reasoning, vol. 49, 2008, pp. 272–284.

[7] J. W. Grzymala-Busse, "Rough set strategies to data with missing attribute values," in Workshop Notes, Foundations and New Directions of Data Mining, in conjunction with the 3-rd International Conference on Data Mining, 2003, pp. 56–63.

[8] ——, "Data with missing attribute values: Generalization of indiscernibility relation and rule induction," Transactions on Rough Sets, vol. 1, 2004, pp. 78–95.

[9] ——, "Generalized parameterized approximations," in Proceedings of the RSKT 2011, the 6-th International Conference on Rough Sets and Knowledge Technology, 2011, pp. 136–145.

[10] P. G. Clark and J. W. Grzymala-Busse, "Experiments on probabilistic approximations," in Proceedings of the 2011 IEEE International Conference on Granular Computing, 2011, pp. 144–149.

[11] J. W. Grzymala-Busse and A. Y. Wang, "Modified algorithms LEM1 and LEM2 for rule induction from data with missing attribute values," in Proceedings of the Fifth International Workshop on Rough Sets and Soft Computing (RSSC'97) at the Third Joint Conference on Information Sciences (JCIS'97), 1997, pp. 69–72.

[12] J. Stefanowski and A. Tsoukias, "Incomplete information tables and rough classification," Computational Intelligence, vol. 17, no. 3, 2001, pp. 545–566.

[13] Z. Pawlak, "Rough sets," International Journal of Computer and Information Sciences, vol. 11, 1982, pp. 341–356.

[14] J. W. Grzymala-Busse, "LERS—a system for learning from examples based on rough sets," in Intelligent Decision Support. Handbook of Applications and Advances of the Rough Set Theory, R. Slowinski, Ed. Dordrecht, Boston, London: Kluwer Academic Publishers, 1992, pp. 3–18.

[15] J. W. Grzymala-Busse and W. Ziarko, "Data mining based on rough sets," in Data Mining: Opportunities and Challenges, J. Wang, Ed. Hershey, PA: Idea Group Publ., 2003, pp. 142–173.

[16] S. K. M. Wong and W. Ziarko, "INFER—an adaptive decision support system based on the probabilistic approximate classification," in Proceedings of the 6-th International Workshop on Expert Systems and their Applications, 1986, pp. 713–726.

[17] W. Ziarko, "Variable precision rough set model," Journal of Computer and System Sciences, vol. 46, no. 1, 1993, pp. 39–59.

[18] J. W. Grzymala-Busse, "A new version of the rule induction system LERS," Fundamenta Informaticae, vol. 31, 1997, pp. 27–39.

[19] ——, "MLEM2: A new algorithm for rule induction from imperfect data," in Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, 2002, pp. 243–250.

Intelligent Technique to Accomplish a Effective Knowledge Retrieval from Distributed Repositories.

Antonio Martín, Carlos León
Department of Electronic Technology
Higher Technical School of Computer Engineering
Sevilla, Spain
toni@us.es, cleon@us.es

***Abstract*— Currently, an enormous quantity of heterogeneous and distributed information is stored in the current digital libraries. Access to these collections poses a serious challenge, however, because present search techniques based on manually annotated metadata and linear replay of material selected by the user do not scale effectively or efficiently to large collections. The Artificial Intelligence and Semantic Web provide a common framework that allows knowledge to be shared and reused. In this paper, we propose a comprehensive approach for discovering information objects in large digital collections based on analysis of recorded semantic metadata in those objects and the application of expert system technologies. We suggest a conceptual architecture for a semantic and intelligent search engine. We concentrate on the critical issue of metadata/ontology-based search. More specifically the objective is investigated from a search perspective possible intelligent infrastructures form constructing decentralized digital libraries where no global schema exists. We have used Case Based-Reasoning methodology to develop a prototype for supporting efficient retrieval knowledge from digital library of Seville University. OntoSDL is a collaborative effort that proposes a new form of interaction between people and Digital Libraries, where the latter are adapted to individuals and their surroundings.**

Keywords-Ontology; Semantic Web; Retrieval; Case-based Reasoning; Digital Library; Knowledge Management.

I. INTRODUCTION

A Digital Library (DL) enables users to interact effectively with information distributed across a network. These network information systems support search and display of items from organized collections. In the historical evolution of digital libraries the mechanisms for retrieval of scientific literature have been particularly important. Traditional search engines treated the information as an ordinary database that manages the contents and positions. The result generated by the current search engines is a list of Web addresses that contain or treat the pattern. The useful information buried under the useless information cannot be discovered. It is disconcerting for the end user. Thus, sometimes it takes a long time to search for needed information.

Although search engines have developed increasingly effective, information overload obstructs precise searches. Despite large investments and efforts have been made, there are still a lot of unsolved problems. There are a lot of researches on applying these new technologies into current DL information retrieval systems, but no research addresses the semantic and Artificial Intelligence (AI) issues from the whole life cycle and architecture point of view [1]. Our work differs from related projects in that we build ontology-based contextual profiles and we introduce an approaches used metadata-based in ontology search and expert systems [2].

We study improving the efficiency of search methods to search a distributed data space like a DL. The objective has focused on creating technologically complex environments in Education, Learning and Teaching in the DL domain. We presented an intelligent approach to develop an efficient semantic search engine. It incorporates semantic Web and AI technologies to enable not only precise location of DL resources but also the automatic or semi-automatic learning [3]. We focus our discussion on case indexing and retrieval strategies to provide an intelligent application in searching area. For this reason we are improving representation by incorporating more metadata in the information representation. Our objective here is thus to contribute to a better knowledge retrieval in the digital libraries field. Our approach for realizing content based both search and retrieval information implies the application of the Case-Based Reasoning (CBR) technology [4].

The contributions are divided into next sections. In the first section, short descriptions of important aspects in DL domain, the research problems and current work in it are reported. Then, we summarize its main components and describe how can interact AI and Semantic Web to improve the search engine. Third section focuses on the ontology design process and provides a general overview about our prototype architecture. Next, we study the CBR framework jColibri and its features for implementing the reasoning process over ontologies [5]. Obviously, our system is a prototype but, nevertheless, it gives a good picture of the on-going activities in this new and important area. Finally, we present conclusions of our ongoing work on the adaptation of the framework and we outline future works.

II. MOTIVATIION AND REQUIREMENTS

In the historical evolution of digital libraries, the mechanisms for retrieval of scientific literature have been particularly important. These network information systems support search and display of items from organized collections. Reuse the knowledge is an important area in DL. The Semantic Web provides a common framework that allows knowledge to be shared and reused across community libraries and semantic searchers [6].

This begets new challenges to docent community and motivates researchers to look for intelligent information retrieval approach and ontologies that search and/or filter information automatically based on some higher level of understanding are required. We make an effort in this direction by investigating techniques that attempt to utilize ontologies to improve effectiveness in information retrieval. Thus, ontologies are seen as key enablers for the Semantic Web. The use of AI and ontologies as a knowledge representation formalism offers many advantages in information retrieval [8]. In our work, we analyzed the relationship between both factors ontologies and AI. We have proposed a method to efficiently search for the target information on a DL network with multiple independent information sources [7].

Seville Digital Library (SDL) is dedicated to the production, maintenance, delivery, and preservation of a wide range of high-quality networked resources for scholars and students at University and elsewhere. The hypothesis is that with a CBR expert system and by incorporating limited semantic knowledge, it is possible to improve the effectiveness of an information retrieval system. In this paper, we study architecture of the search layer in this particular dominium, a web-based catalogue for the University of Seville. SDL provides tools that support the construction of online information services for research, teaching, and learning. SDL include services to effectively share their materials and provide greater access to digital content. Our objective here is thus to contribute to a better knowledge retrieval in the digital libraries field.

III. The System Architecture

In order to support semantic retrieval knowledge in a DL, we develop a prototype named OntoSDL based on ontologies and expert systems. The architecture of our system is shown in Fig.1, which mainly includes three parts: intelligent user interface, ontology knowledge base, and the search engine. Their corresponding characteristics and functions are studied in the following paragraphs.

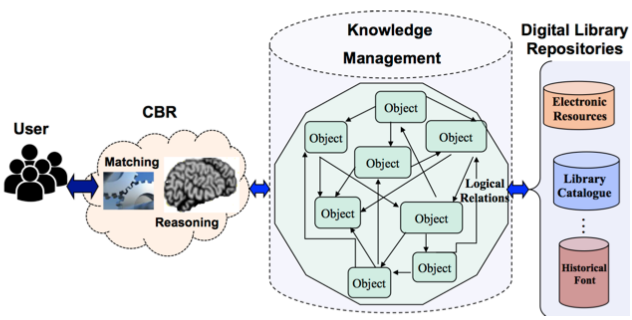


Figure 1. System architecture of OntoSDL

OntoSDL system uses its internal knowledge bases and inference mechanisms to process information about the electronic resources in a DL. At this stage we consider to use ontology as vocabulary for defining the case structure like attribute-value pairs. Ontology will be considered as knowledge structure that will identify the concepts, property of concept, resources, and relationships among them to enable share and reuse of knowledge that are needed to acquire knowledge in a specific search domain.

Ontology knowledge base is the kernel part for semantic retrieval information. The metadata descriptions of the resources and library objects (cases) are abstracted from the details of their physical representation and are stored in the case base. Ontology stores information about resources and services where concepts are types, or classes, individuals are allowed values, or objects and relations are the attributes describing the objects.

Inference engine contains a CBR component that automatically searches for similar queries-answer pairs based on the knowledge that the system extracted from the questions text. Case base has a memory organization interface that assumes that whole case-base can be read into memory for the CBR to work with it. We used a CBR shell, software that can be used to develop several applications that require cased-based reasoning methodology. Also we have implemented a new interface, which allows retrieving cases enough to satisfy a SQL query. In this work, we analyzed the CBR object-oriented framework development environments JColibri. This framework work as open software development environment and facilitate the reuse of their design, as well as implementations.

The acceptability of a system depends to a great extent on the quality of his user interface component. In our system, the user interacts with the system to fill in the gaps to retrieve the right cases. The interfaces provide for browsing, searching and facilitating Web contents and services. It consists of one user profile, consumer search agent components and bring together a variety of necessary information from different user's resources. The user interface helps to user to build a particular profile that contains his interest search areas in the DL domain. The objective of profile intelligence has focused on creating of user profiles: Staff, Alumni, Administrator, and Visitor.

We have developed a graphical selection interface as illustrated in Fig. 2.

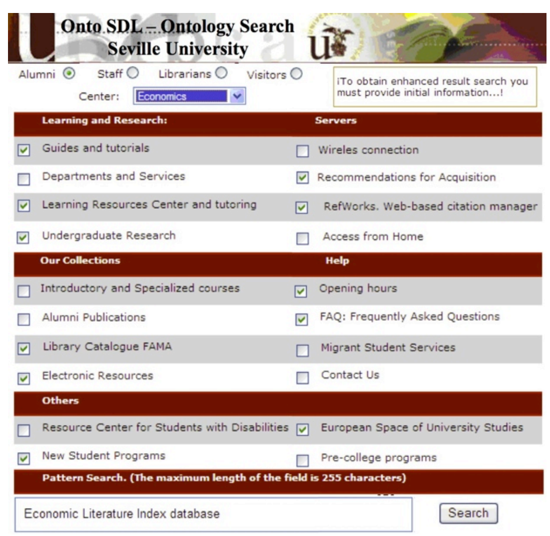


Figure 2. User profiles interface

In an intelligence profile setting, people are surrounded by intelligent interfaces merged. Rather than building static user profiles, contextual systems try to adapt to the user's current search. OntoSDL monitors user's tasks, anticipates search-based information needs, and proactively provide users with relevant information. Thus creating a computing-capable environment with intelligent communication and processing available to the user by means of a simple, natural, and effortless human-system interaction. The user enters query commands and the system asks questions during the inference process. Besides, the user will be able to solve new searches for which he has not been instructed, because the user profiles what he has learnt.

IV. CASE-BASED REASONING INTELLIGENT TECHNIQUE

CBR is widely discussed in the literature as a technology for building information systems to support knowledge management, where metadata descriptions for characterizing knowledge items are used. CBR is a problem solving paradigm that solves a new problem, in our case a new search, by remembering a previous similar situation and by reusing information and knowledge of that situation. A new problem is solved by retrieving one or more previously experienced cases, reusing the case, revising. In our CBR application, problems are described by metadata concerning desired characteristics of a library resource, and the result to a specific search is a pointer to a resource described by metadata. These characterizations are called cases and are stored in a case base. CBR case data could be considered as a portion of the knowledge (metadata) about an OntoSDL object. Every case contains both a solution pointers and problem description used for similarity assessment. Description of the framework case, which is formally described in terms of framework domain taxonomy they are used for indexing cases. The possible solutions described by means of framework instantiation actions and additional information to justifies these steps. The following processes may describe a CBR cycle, Fig. 3:

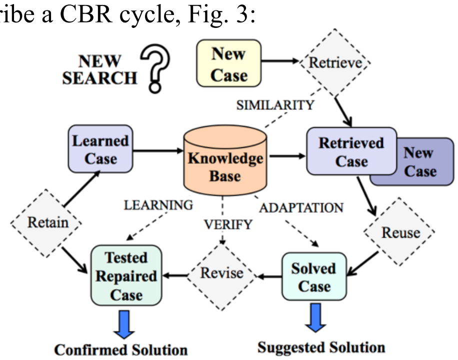


Figure 3. User profiles interface

- Retrieval: main focus of methods in this category is to find similarity between cases. Similarity function can be parameterized through system configuration.
- Reuse: a complete design where case-based and slot-based adaptation can be hooked is provided.
- Revise the proposed solution if necessary. Since the proposed result could be inadequate, this process can correct the first proposed solution.
- Retain the new solution as a part of a new case. This process enables CBR to learn and create a new solution that should be added to the knowledge base.

A. *CBR Structure*

The development of a quite simple CBR application already involves a number of steps, such as collecting case and background knowledge, modeling a suitable case representation, defining an accurate similarity measure, implementing retrieval functionality, and implementing user interfaces. Compared with other AI approaches, CBR allows to reduce the effort required for knowledge acquisition and representation significantly, which is certainly one of the major reasons for the commercial success of CBR applications. Nevertheless, implementing a CBR application from scratch remains a time-consuming software engineering process and requires a lot of specific experience beyond pure programming skills.

Although CBR claims to reduce the effort required for developing knowledge-based systems substantially compared with more traditional AI approaches. The implementation of a CBR application from scratch is still a time consuming task. We present a novel, freely available tool for rapid prototyping of CBR applications. CBR object-oriented framework development environments JColibri have been used in this study. By providing easy to use model generation, data import, similarity modeling, explanation, and testing functionality together with comfortable graphical user interfaces, the tool enables even CBR novices to rapidly create their first CBR applications. Nevertheless, at the same time it ensures enough flexibility to enable expert users to implement advanced CBR applications [9].

jColibri is and open source framework and their interface layer provides several graphical tools that help users in the configuration of a new CBR system. Our motivation for choosing this framework is based on a comparative analysis between it and other frameworks, designed to facilitate the development of CBR applications. jColibri enhances the other CBR shells: CATCBR, CBR*Tools, IUCBRF, Orenge. Another decision criterion for our choice is the easy ontologies integration. jColibri affords the opportunity to incorporate ontology in the CBR application to use it for case representation and content-based reasoning methods to assess the similarity between them.

B. *Retrieval of similar cases process*

The main purpose of establishing intelligent retrieval ontology is to provide consistent and explicit metadata in the process of knowledge retrieval. CBR systems typically apply retrieval and matching algorithms to a case base of past search-result pairs. CBR is based on the intuition that new searches are often similar to previously encountered searches, and therefore, that past results may be reused directly or through adaptation in the current situation. Our system provides multilayer retrieval methods:

1. Intelligent profiles interface: Low-level selection of query profile options, which mainly include the four kinds of user. These users can specify certain initial items, i.e., the characteristics and conditions for a search. For this a statistical analysis has been done to determine the importance values and establishing specified user requirements. User searches are monitored by capturing information from different user profiles. This statistical analysis even can in fact lay the foundation for searches in a particular user profile.

2. Ontology semantic search can query on classes, subclasses or attributes of knowledge base, and matched cases are called back.

3. The retrieval process identifies the features of the case with the most similar query. Our inference engine contains the CBR component that automatically searches for similar queries-answer pairs based on the knowledge that the system extracted from the questions text. The system uses similarity metrics to find the best matching case. Similarity measures used in CBR are of critical importance during the retrieval of knowledge items for a new query. Similarity retrieval expands the original query conditions, and generates extended query conditions, which can be directly used in knowledge retrieval. Unlike in early CBR approaches, the recent view is that similarity is usually not just an arbitrary distance measure, but function that approximately measures utility.

We used a computational based retrieval where numerical similarity functions are used to assess and order the cases regarding the query. The retrieval strategy used in our system is nearest-neighbor approach. This approach involves the assessment of similarity between stored cases and the new input case, based on matching a weighted sum of features. A typical algorithm for calculating nearest neighbor matching is next:

$$similarity(Case_I, Case_R) = \frac{\sum_{i=1}^{n} w_i \times sim(f_i^I, f_i^R)}{\sum_{i=1}^{n} w_i} \quad (1)$$

Where w_i is the importance weighting of a feature (or slot), sim is the similarity function of features, and f_i^I and f_i^R are the values for feature i in the input and retrieved cases respectively.

The use of structured representations of cases requires approaches for similarity assessment that allow to compares two differently structured objects, in particular, objects belonging to different object classes. An important advantage of similarity-cased retrieval is that if there is no case that exactly matches the user's requirements, this can show the cases that are most similar to his query.

V. ONTOLOGY DESIGN AND DEVELOPMENT

We need a vocabulary of concepts, resources, and services for the knowledge system described. This scenario requires definitions about the relationships between objects of discourse and their attributes [10]. We have proposed to use ontology together with CBR in the acquisition of the knowledge in the specific DL domain. The primary information managed in the OntoSDL domain is metadata about library resources, such as books, digital services and resources, etc. We integrated three essential sources to the system: electronic resources, catalogue, and personal Data Base.

The W3C defines standards that can be used to design an ontology [11]. We wrote the description of these classes and the properties in RDF semantic markup language. RDF is used to define the structure of the metadata describing DL resources. OntoSDL project contains a collection of codes, visualization tools, computing resources, and data sets distributed across the grids, for which we have developed a well-defined ontology using RDF language. Our ontology can be regarded as triplet OntoSearch:={profile, collection, source) where profiles represent the user kinds, collection contains all the services and sources of the DL, and source cover the different information root: catalogue, history fond, intranet, Web, etc. We choose Protégé as our ontology editor, which supports knowledge acquisition and knowledge base development [12]. It is a powerful development and knowledge-modeling tool with an open architecture. Protégé uses OWL and RDF as ontology language to establish semantic relations.

In order to realize ontology-based intelligent retrieval, we need to build case base of knowledge with inheritance structure. The ontology and its sub-classes are established according to the taxonomies profile, as shown in Fig. 4.

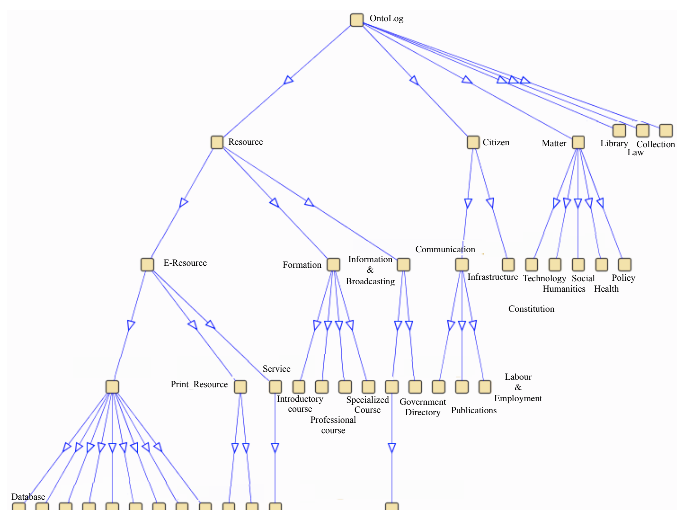


Figure 4. Class hierarchy for the OntoSDL ontology

This shows the high level classification of classes to group together OntoSDL resources as well as things that are related with these resources. As shown the Fig. 4, profile ontology includes several attributes like Electronic_Resources, Digital_Collections, Catalogue, Science_Resources, etc. After ontology is established, we need to add enough initial instances and item instances to knowledge base. For this purpose we followed these steps:

 ISBN: 978-1-61208-352-0

first we choose a certain item, and create a blank instance for item; second the domain expert, in this case the librarian fills blank units of instance according the domain knowledge. To finish, the library of cases (the "case base") is generated from a file store where each case is represented with RDF syntax.

1100 cases were collected for user profiles and their different resources and services. This is sufficient for our proof-of-concept demonstration, but would not be sufficiently efficient to access large resource sets. Each case contains a set of attributes concerning both metadata and knowledge. However, our prototype is currently being extended to enable efficient retrieval directly from a database, which will enable its use for large-scale sets of resources.

VI. EXPERIMENTAL EVALUATION

Experiments have been carried out in order to test the efficiency of AI and ontologies in retrieval information in a DL. These are conducted to evaluate the effectiveness of run-time ontology mapping. The main goal has been to check if the mechanism of query formulation, assisted by an agent, gives a suitable tool for augmenting the number of significant documents, extracted from the DL to be stored in the CBR. The user begins the search devising the starting query. Suppose the user is looking for some resource about "Computer Science electronic resource" in the library digital domain of Seville, Fig. 5.

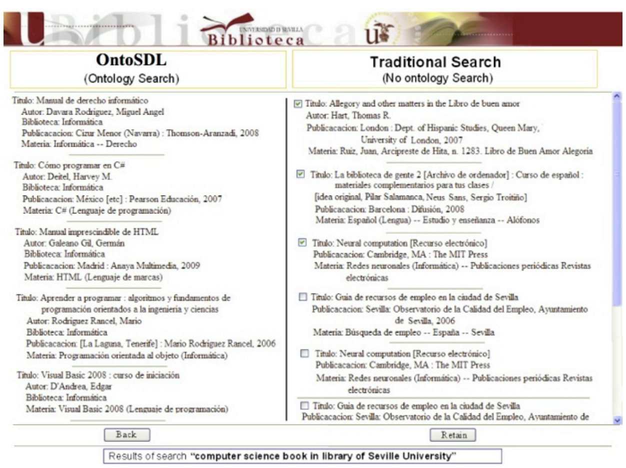


Figure 5. Search engine results page

The user inputs the keywords in the user profile interface. The required resources should contain some knowledge about "Computer Science" and related issues. After searching, some resources are returned as results. The results include a list of web pages with titles, a link to the page, and a short description showing where the keywords have matched content within the page.

We have compared our prototype with some semantic search engines like Hakia, Lexxe, SenseBot, etc. However, we have focused in Google because is the world's dominant search engine and Google has made significant inroads in semantic indexing in search. It is a fact that deep inside Google is based on breakthrough semantic search techniques that are transforming Google's search results [13].

For our experiments we considered 50 users with different profiles. Therefore, we could establish a context for the users, they were asked to at least start their essay before issuing any queries to OntoSDL. They were also asked to look through all the results returned by OntoSDL before clicking on any result. We compared the top 10 search results of each keyword phrase per search engine. Our application recorded which results on which they clicked, which we used as a form of implicit user relevance in our analysis. We must consider that retrieved documents relevance is subjective. That is different people can assign distinct values of relevance to a same document. In our study, we have agreed different values to measure the quality of retrieved documents, excellent, good, acceptable and poor.

In each experiment, we report the average rank of the user-clicked result for our baseline system, Google and for our search engine OntoSDL. Next, we calculated the rank for each retrieval document by combining the various values and comparing the total number of extracted documents and documents consulted by the user (Table 1).

TABLE I. ANALYSIS OF RETRIEVED DOCUMENTS RELEVANCE FOR SELECT QUERIES

	Excellent	Good	Acceptable	Poor
OntoSDL	7,50%	41,50%	40,60%	10,40%
Google	2,60%	27,90%	43,40%	26,10%

After the data was collected, we had a log of queries averaging 5 queries per user. Of these queries, some of them had to be removed, either because there were multiple results clicked, no results clicked, or there was no information available for that particular query. The remaining queries were analyzed and evaluated. These results are presented in Fig. 6.

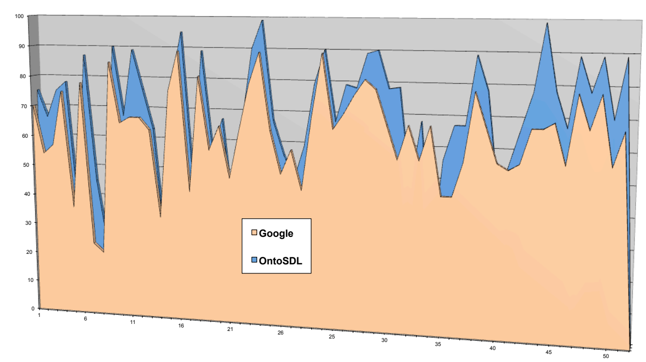


Figure 6. Search engine results page

In our study DL domain we can observe the best final ranking was obtained for our prototype OntoSDL and an interesting improvement over the performance of Google. Test of significance is the analysis of the number of searches that have been resolved satisfactory by OntoSDL. As noted in Table 1 our system performs satisfactorily with about a 91.6% rate of success in real cases.

Another important aspect of the design and implementation of an intelligent system is determination of the degree of speed in the answer that the system provides. During the experimentation, heuristics and measures that are commonly adopted in information retrieval have been used. While the users were performing these searches, an application was continually running in the background on the server, and capturing the content of queries typed and the results of the searches. Statistical analysis has been done to determine the importance values in the results. We can establish that speed in our system improves the proceeding time and the average of the traditional search engine. The results for OntoSDL are 9.15% better than proceeding time and 11.9% better than executing time searches/sec in the traditional search engines.

VII. CONCLUSION AND FUTURE WORK

We have investigated how semantic technologies and AI can be used to provide additional semantics from existing resources in digital libraries. We described an effort to design and develop a prototype for management the resources in a library such as OntoSDL project, and to exploit them to aid users as they select resources. Our study addresses the main aspects of a Semantic Web knowledge retrieval system architecture trying to answer the requirements of the next-generation Semantic Web user. This scheme is based on the next principle: knowledge items are abstracted to a characterization by metadata description and it is used for further processing.

For this purpose we presented a system based in ontology and AI architecture for knowledge management in the Seville DL. To put our aims into practice we should first of all develop the domain ontology and study how the content-based similarity between the concepts typed attributes could be assessed in CBR system. A dedicated inference mechanism is used to answer queries conforming to the logic formalism and terms defined in our ontology. We have been working on the design of entirely ontology-based structure of the case and the development of our own reasoning methods in jColibri to operate with it. It introduced a prototype web-based CBR retrieval system, which operates on an RDF file store. Furthermore an intelligent agent was illustrated for assisting the user by suggesting improved ways to query the system on the ground of the resources in a DL according to his own preferences, which come to represent his interests.

Finally, the study analyzes the implementation results, and evaluates the viability of our approaches in enabling search in intelligent-based digital libraries. The results demonstrate that by improving representation by incorporating more metadata from within the information and the ontology into the retrieval process, the effectiveness of the information retrieval is enhanced. Future work will concern the exploitation of information coming from others libraries and services and further refine the suggested queries, to extend the system to provide another type of support, as well as to refine and evaluate the system through user testing. It is also necessary the development of an authoring tool for user authentication, efficient ontology parsing and real-life applications.

REFERENCES

[1] D. Govedarova, S. Stoyanov, and I. Popchev, “An Ontology Based CBR Architecture for Knowledge Management in BULCHINO Catalogue” International Conference on Computer Systems and Technologies”, 2008.

[2] P. Warren. “Applying semantic technologies to a digital library: a case study”, "Applying semantic technology to a digital library: a case study", Library Management, Vol. 26 Iss: 4/5, pp.196 – 205, 2005.

[3] H. Stuckenschmidt and F. Harmelen, “Ontology-based metadata generation from semi-structured information”, K-CAP '01: Proceedings of the 1st international conference on Knowledge capture, 2001, pp. 163-170, doi:10.1145/500737.500763

[4] J. Toussaint and K. Cheng, “Web-based CBR (case-based reasoning) as a tool with the application to tooling selection”, The International Journal of Advanced Manufacturing Technology, 29(1-2): pp. 24 - 34, May 2006, DOI: dx.doi.org/10.1007/s00170-004-2501-0

[5] GAIA - Group for Artificial Intelligence Applications. “*jCOLIBRI project - Distribution of the development environment with LGPL”*, http://gaia.fdi.ucm.es/grupo/projects/,Complutense University of Madrid, January, 2014.

[6] Y. Sure and R. Studer, “Semantic web technologies for digital libraries”, Library Management Journal, Emerald, Library Management, Vol. 26 Iss: 4/5, 2005, pp.190 - 195

[7] H. Ding, “Towards the metadata integration issues in peer-to-peer based digital libraries”, GCC (H. Jin, Y. Pan, N. Xiao, and J. Sun, eds.), vol. 3251 of Lecture Notes in Computer Science, Springer, 2004, pp 851-854.

[8] M. Bridge, H. G¨oker, L McGinty, and B. Smyth, “Case-based recommender systems”, The Knowledge Engineering Review archive, Volume 20 Issue 3, September 2005, Pages 315 - 320, doi>10.1017/S0269888906000567

[9] B. Díaz-Agudo, P.A. González-Calero, J.Recio-García, and A Sánchez-Ruiz, “Building CBR systems with jColibri”, Journal of Science of Computer Programming, Volume 69, Issues 1–3, 1 December 2007, Pages 68–75, doi: dx.doi.org/10.1016/j.scico.2007.02.004.

[10] S. Staab and R. Studer, “Handbook on Ontologies. International Handbooks on Information Systems”, Springer, Berlin, 2005.

[11] W3C, *RDF Vocabulary Description Language 1.0: RDF Schema,* http://www.w3.org/TR/rdf-schema/, January, 2014.

[12] PROTÉGÉ, *The Protégé Ontology Editor and Knowledge Acquisition System,* <http://protege.stanford.edu/>, January, 2014.

[13] D. Amerland, “Google Semantic Search: Search Engine Optimization (SEO) Techniques That Get Your Company More Traffic, Increase Brand Impact and Amplify Your Online Presence, Que Publishing Kindle Edition, July, 2013.

Self-managed Crowdsourcing

Vladislav Protasov, Zinaida Potapova, Eugene Melnikov
Center of Computing for Physics and Technology
Moscow, Russia
protvlad@gmail.com, zinaida.potapova@gmail.com, apinae1@gmail.com

***Abstract*—The paper presents a technology of crowdsourcing organization that is considered to provide optimal solutions of complicated intelligent problems from large group of experts. This approach is developed as an extension of the Evolutionary Solutions Coordination method; it supposes dividing entire expert group into small ones and automatically defining best experts on each iteration to form the most competent groups on further iterations. Finally, the group of the most competent experts gives full and the most rational solution. The paper also presents an experiment on the Eysenck's tests solving. It demonstrates significant superiority of self-managed crowdsourcing over individual solution approach.**

Keywords—crowdsourcing; collective intelligence; coordination; generation; estimation; election; social.

I. INTRODUCTION

Growth of the Internet and evolution of web technologies gave birth to a new kind of collective interaction of the web users – crowdsourcing, which means creating needed solutions, ideas and other intellectual products with contribution of large number of people, usually Internet users. This term was first used by the journalist Jeff Howe [1]. In Russia, this technology is developed by a young company Witology performed a set of considerable projects for Russian economy [2]. Unfortunately, according to experience of crowdsourcing practice, the technology is rather spendy, it requires usage of expensive software, large calculation resources and numerous project coordinators, so called facilitators [3], it makes the technology almost unreachable for small companies and certain Internet users, who also may represent interesting social projects.

The technology of Self-managed Crowdsourcing presented in this paper does not have these disadvantages. It utilizes a special algorithm that is able to coordinate collective solution search automatically, without human intervention [4].

The first section of the paper deals with theoretical basics of the presented algorithm. It explains the logic of three-cycle scheme step by step. In the next section the theory of self-managed crowdsourcing is presented. After that some experimental results will be provided. The experiment considers a group of students that solve IQ tests using the provided theory. The last section gives a conclusion upon the described experiment.

II. THREE-CYCLE SCHEME OF CROWDSOURCING

Papers on crowdsourcing state that group solution can significantly overcome individual results if there is good algorithmic platform of collective work organization [5]. Method of Evolutionary Solutions Coordination described by Protasov et al. in [6] can be used as a kind of crowdsourcing approach on large experts number. Therefore, there is a need to develop such a collaboration technology that leads the number of coordination steps to minimum.

According to the procedure of Evolutionary Solutions Coordination, in order to minimize coordination steps number, the following organization of self-managed crowdsourcing can be considered. On the first cycle, each expert fills slots of the projects with his own proposals. On the second cycle, basing on predefined regular graph [7] of connections experts receive others' solutions and fill up empty slots of their own solutions with received variants that seem correct in their view [8]. On the third cycle, for each slot of the project, the value chosen by greater than a half of experts is considered as the group's choice. The group's solution is formed of such slots' values.

Of course, on the first cycle, it is desirable to provide equiprobable slots filling. Regular graph is applicable for this task because there is only iteration of solutions coordination and random genetic-like expert pairs selection is not necessary. It is also noticeable that each cycle is performed by all experts concurrently and the operation time does not depend on their quantity.

The obtained results are also significant for the artificial intelligence construction of large number of uniform modules, e. g., neurons cluster or primitive computers forming homogeneous environment.

The scheme of three-cycle coordination is illustrated on Fig. 1.

III. THE TECHNOLOGY OF SELF-MANAGED CROWDSOURCING

Let us consider the Evolutional Coordination Method application based on social web-platform – an Internet site constructed specially for organizing crowdsourcing technology. The number of experts is not defined previously. Collective work can be described as follows. The experts registered on the site form small groups, e. g., 10 people each group. Each expert is provided with a predefined project task and an instruction which contains his groupmate list for sending initial ideas and time interval allotted for ideas generation. After finishing the first stage and receiving other users' solutions for expertise, the experts perform an estimation of the received solutions. All slots' values which seem correct for them are copied to their own solutions. After the received variants estimation and slots filling, all solutions are saved on the site. Then, the special program, Project Moderator, picks out best

 ISBN: 978-1-61208-352-0

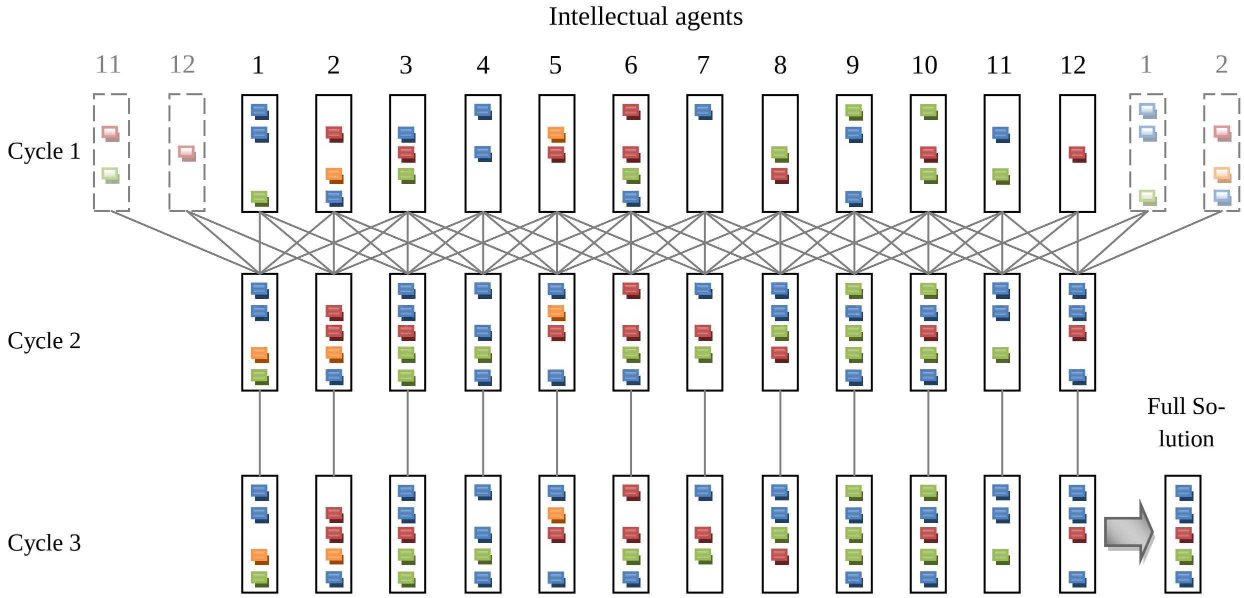


Figure 1. Three-cycle coordination scheme

experts (ones with greatest estimation ability in a group) from all groups and allocates them into new groups. These groups are given the best variants of previous expert generation and estimation cycle is repeated. Further, the best experts are selected again from these groups; the process proceeds until the best final group is selected among experts, and they obtain the last and the most fully defined solution.

Such scheme of crowdsourcing organization induces self-managed processes of the best experts distinguishing and obtaining the best solution with small number of iterations. Existing crowdsourcing schemes widely utilize such technology with inviting large groups of human moderators. This makes projects very expensive. But, the technology described in this paper supposes to entrust the coordination work to a program.

IV. EXPERIMENTAL TESTING OF SELF-MANAGED CROWDSOURCING TECHNOLOGY

In order to test the self-manged crowdsourcing technology, some experiments in Internet were carried out. 7 groups of students consisting of 3 to 7 people participated in the experiments. Eysenck intellect testing with predefined test examples was selected as a test task [9]. It included 50 questions where the answers were single keywords. The experiment was hold according to the crowdsourcing technology described above. Students registered beforehand on the site of the project and were united into groups of 3, 4, 5, 6, 7, 5 and 7 people (see the Table 2). According to the Eysenck methodology each group was given 30 minutes to complete the test, the time intervals allocated for the first and second cycles were defined depending on a group size: 18 minutes for 7 and 24 minutes for 3 people for the first cycle – the time varies because more experts need more time for coordination stage.

The table contains competence data of all experts: slots number filled on first and second cycles, competence in generation and estimation, forecast of correct answers number (N), IQ level calculated by the Eysenck's formula $\mathrm{IQ} = 90 + 2.5N$ and the Intellectual Potential (IP), where IP is defined by the formula

$$\mathrm{IP} = \frac{1000}{M} \tag{1}$$

where M, for certain expert, is the number of experts with the same competence (both generative and estimative) that provides full solution with probability 0.999.

Analyzing the table, it's easy to descry that IP distribution has considerably greater dispersion then that of IQ level. In this example it vary from 1.5 to 144.7 while IQs of corresponding experts equal 104 and 165. Since IP shows more correctly the certain expert's contribution to overall work, it can be deduced that estimation of experts only by IQ gives undeserved advantage to less smart ones. Table 1 illustrates full results of expert groups work. It contains a number of slots filled on the second cycle, portion of filled slots after first and second cycles and IQ level for each group.

TABLE 1. EXPERT GROUPS RESULTS

Group N	Slots filled on 2st cycle	Relative slots filling on 1st cycle	Relative slots filling on 2st cycle	IQ
1	19	0,06	0,38	138
2	24	0,18	0,48	150
3	28	0,06	0,56	160
4	25	0,02	0,5	152
5	14	0,02	0,28	125
6	21	0,06	0,42	142
7	23	0	0,46	148
3rd cycle	50	0,52	1	215

 ISBN: 978-1-61208-352-0

TABLE 2. EXPERTS RATINGS

Group *N*	Expert *N*	Slots filled on 1st cycle	Slots filled on 2st cycle	Generation competence	Estimation competence	Correct answers forecast (N)	IQ	IP
1	1	3	26	0,075	0,74	3,75	99	11,3
	2	18	19	0,45	0,0625	22,5	146	30,2
	3	19	21	0,475	0,133	23,75	149	53,2
2	1	6	23	0,156	0,654	7,82	110	23,8
	2	6	0	0,156	0,01	7,82	110	1,9
	3	14	20	0,365	0,333	18,3	136	54,4
	4	14	22	0,365	0,555	18,3	136	63,1
3	1	11	37	0,3	1	15	128	55,1
	2	8	28	0,218	0,69	10,9	117	35,1
	3	1	16	0,02	0,432	1	93	2,9
	4	3	27	0,082	0,706	4,09	100	12,3
	5	13	28	0,355	0,625	17,7	134	62,5
4	1	6	25	0,171	0,543	8,57	111	25,1
	2	1	29	0,02	0,7	1	93	3,2
	3	14	30	0,4	0,222	20	140	52,7
	4	5	6	0,143	0,028	7,14	108	3,9
	5	4	0	0,115	0,01	5,71	104	1,5
	6	11	38	0,314	0,11	15,7	129	26,1
5	1	4	12	0,12	0,348	6	105	14,9
	2	6	11	0,18	0,238	9	113	19,9
	3	11	0	0,33	0,01	16,5	131	4,3
	4	11	13	0,33	0,125	16,5	131	30,1
	5	2	11	0,06	0,36	3	98	7,5
	6	11	21	0,33	0,625	16,5	131	56,8
	7	7	20	0,21	0,65	10,5	116	33,1
6	1	5	19	0,136	0,634	6,82	107	20,3
	2	7	25	0,191	0,9	9,54	114	31,8
	3	9	19	0,245	0,556	12,3	121	38,2
	4	11	14	0,3	0,187	15	128	32,9
	5	10	24	0,273	0,823	13,6	124	47,4
7	1	3	4	0,094	0,025	4,74	102	2,4
	2	4	0	0,126	0,01	6,31	106	1,6
	3	10	13	0,316	0,091	15,8	130	23,3
	4	19	41	0,6	0,917	30	165	144,7
	5	6	28	0,189	0,594	9,47	114	28,8
	6	6	11	0,189	0,135	9,47	114	15,8
	7	12	14	0,379	0,064	18,9	137	23,6

According to the technology of Self-managed Crowdsourcing, on final cycle the group of experts with the best competence in solutions estimation was selected from the whole set of experts. They interchanged their solutions and complemented each other. After the coordination cycle, for the final solution, the slots presented in most experts' solutions were distinguished to form the final solution. The results are presented in the last row of the Table 1. On the last cycle of the experiment selected group of the best experts obtained full solution – all questions of the test were answered.

V. CONCLUSION AND FUTURE WORK

Finally, the first cycle of the Self-managed Crowdsourcing includes the creation of population of correctly filled slots. After second cycle, many experts experience considerable growth of correctly filled slots, but most experts correct answers portion is still low and the group has not yet filled all the slots of the projects. Only small group of leaders has estimative ability greater than 0.5, they actually have overcome the Condorcet's border [10]. These very leaders formed the final group that obtained full solution on the last cycle. Taking into account that Eysenck's tests are constructed with consideration of impossibility to solve them fully in given time it can be concluded that expert groups under Self-managed Crowdsourcing technology have done an impossible.

ACKNOWLEDGEMENTS

This work was supported by the Russian Foundation for Basic Research, grant #13-07-00958 "Development of the theory and experimental research of a new information technology of self-managed crowdsourcing".

REFERENCES

[1] Howe J., The Rise of Crowdsourcing, www.wired.com, [retrieved: 04, 2014], p. 1.
[2] Witology, the first Russian crowdsourcing company, www.witology.com/company, [retrieved: 04, 2014].
[3] Shih-Wen Huang, Wai-Tat Fu, "Motivating Crowds Using Social Facilitation and Social Transparency", CHI 2011, Vancouver BC Canada, May 2011, p. 2.
[4] Protasov V. I. and Potapova Z. E., "Self-Governing Crowdsourcing. Theory, Technology and Practice", The 3rd International Conference on Control Automation, Intelligent Systems and Environments, Makhachkala, 2012, RAS Kabardino-Balkar Science Center Press, vol. 1, pp. 91-99.
[5] Protasov V., Vitiska N., and Shelhkova L., "Use of collective intelligence of group of robots for acceleration of acceptance expedient decisions", Conference "Intellectual robotic systems", Russia, Gelendjik, Oct. 2001, pp. 187— 189.
[6] Protasov V., Potapova Z., and Melnikov E., "Overcoming the Condorcet's Border in Collective Intelligence Systems", The Second International Conference on Intelligent Systems and Applications INTELLI, 2013 April 21 - 26, 2013 - Venice, Italy, pp. 2-4.
[7] Regular Graph, en.wikipedia.org/wiki/Regular_graph [retrieved: 04, 2014], p. 1
[8] Protasov V., Vitiska N., and Shustov E., "Solving complex problems with a genetic consilium method", Conference "Intellectual robotic systems", Russia, Gelendjik, Oct. 2001, pp. 53-55.
[9] Eysenck H. J., "The Structure and Measurement of Intelligence. Tests IQ", University of London. Institute of Psychiatry, 2001, p. 128.
[10] Le Marquis de Condorcet, "Essay on the Application of Analysis to the Probability of Majority Decisions", 1785, Les Archives de la Revolution Française, Pergamon Press, pp. 9-23.
[11] Turchin V., "The phenomenon of science. A cybernetic approach to human evolution", Columbia University Press, New York, 1977, pp. 245-250.
[12] Protasov V. I., "Design of Metasystem Tansitions", Physical and Technical Informatics Institute Press, Moscow, 2009; p. 186.
[13] Protasov V., "Generation of New Knowledge by Network Human-machine Intelligence. Statement of a problem.", J. Neurocomputers. Development and application, vol.7-8, Moscow, 2001, pp. 94 – 103.

 ISBN: 978-1-61208-352-0

Refining the Scatteredness of Classes using Pheromone-Based Kohonen Self-Organizing Map (PKSOM)

Azlin Ahmad
Faculty of Computer and Mathematical Sciences
Universiti Teknologi MARA (UiTM)
Shah Alam, Malaysia
Center of Artificial Intelligence and Robotics (CAIRO),
Malaysia Japan International Institute of Technology (MJIIT)
Universiti Teknologi Malaysia (UTM)
Kuala Lumpur, Malaysia
azlin@tmsk.uitm.edu.my

Rubiyah Yusof
Center of Artificial Intelligence and Robotics (CAIRO),
Malaysia Japan International Institute of Technology (MJIIT)
Universiti Teknologi Malaysia (UTM)
Kuala Lumpur, Malaysia
rubiyah@ic.utm.my

***Abstract*— The Kohonen Self-Organizing Map (KSOM) is one of the well-known unsupervised learning algorithms, which has been applied in various areas. This algorithm can cluster and classify an enormous amount of data into several clusters according to the similarity of the data features. However, it has many drawbacks, such as difficulty of clustering the data, which have similar features. These may lead to the inefficient result; the data is scatteredly mapped even though it is accurately clustered into several clusters according to the features. Therefore, this paper proposed a Pheromone-based Kohonen Self-Organizing Map (PKSOM) algorithm to refine the scatteredness of the data in the clusters, thus to improve the cluster density. Some modifications have been made to the original Kohonen Self-Organizing Map (KSOM), adapted from the Ant Clustering Algorithm procedures. This PKSOM has been tested on three different datasets; Iris flowers, Glass and Wood datasets. Based on the result, the proposed method has improved the classification impressively by increasing the density of the data in clusters. Hence, it has also refined the scatteredness of classes, where each dataset is well clustered where data that have similar features are located closely to each other in the same cluster. However, there are a few overlapped clusters that still occur.**

Keywords–Kohonen Self-Organizing Map (KSOM); Pheromone; Ant Clustering Algorithm (ACA); Clustering; Cluster density.

I. INTRODUCTION

The Kohonen Self-Organization Map (KSOM) is an unsupervised learning technique that is expended by topological self-organizing maps stemming from techniques that were first proposed for competitive learning [1]. This algorithm is used to process the high dimensional data, and it is also designed to cluster the data into clusters of data that exhibit some similarities. Each cluster with similar features is projected onto the same node on the map. Otherwise, the dissimilarity increases with the distance that separates two projections on the map. Thus, the cluster space is identified to the map, so that the projection enables simultaneous visualization of the cluster and the observation space [2]. Being one of the most popular unsupervised learning technique, KSOM has been used in different areas, in clustering, that might help to solve a complex problem. Giraudel et al. [3] discussed a comparison between the application of KSOM and other conventional ordination methods for ecological community. As mentioned earlier, KSOM has been used in different areas for many purposes. Anthony has used the KSOM to develop a new color quantization algorithm for mapping the 24-bit color images to eight-bit color [4]. They proved that their proposed algorithm could produce better output compared to the Oct-Tree and Median-Cut algorithms with very limited samples. While Emamian et al. [5] have proposed the application of KSOM to recognize the transient crack-related signals in the presence of strong time-varying noise and other interferences. The application can cluster the acoustic emission signals for fault monitoring and as a result, it is showed that the KSOM did perform well with a small probability of error.

The KSOM has also been widely used as a visualization tool for dimensionality reduction. Its unique topology preserving property can be used to visualize the relative mutual relationships among the data. It has been applied to organize and visualize vast amount of textual information, for example, the SOM that organizes massive document collection; WEBSOM [6]. The main benefit of KSOM is the topology preservation of an input space, which makes similar object appear closely on the map. Most of these applications, however, are based on 2D grids and map. Weijian et al. [7] investigated a hybrid neural network framework by combining the supervised learning algorithm with unsupervised algorithm on integrated representation platform of multiple two dimensional KSOM with the assistance of associative memory for clustering and classification of Remotely Sensed (RS) imagery. The formation of the clusters and the transformation from clusters to decision regions are implemented by unsupervised and supervised self-organizing learning on several Kohonen 2D Self-Organizing Maps (M2dSOM), individually. Xu et al. [8] used the two important operations in KSOM: vector quantization and topological preserving mapping, while introducing an online Self-Organizing Topological Tree (SOTT), with faster learning, is proposed. Their proposed learning rule is novel and delivers the efficiency and the topological preservation compared to other structures of KSOM. The computational complexity of

 ISBN: 978-1-61208-352-0

SOTT is better and its computation time is much shorter than the entire search process done by KSOM. Forkan et al. [9] proposed a new method for surface based on hybrid techniques using KSOM and Particle Swarm Optimization (PSO). The KSOM learns the sample data through mapping grid. Consequently, the learned and well-represented data has become the input for surface fitting procedure. The authors have proposed PSO to probe the optimum fitting points on surfaces and this algorithm has been applied on different types of curve to observe its ability in reconstructing the object while preserving the original shapes.

However, the KSOM has several weaknesses, such as the difficulty of clustering and classifying the data, which have similar features, and this may lead to an inefficient result. Therefore, to improve the clustering result, a few researchers have proposed an optimized Kohonen by hybridizing KSOM with other techniques such as K-mean [19], simulated annealing [20], rough set theory and genetic algorithm [21], and Ant Colony Optimization (ACO) [22][23].

There are numerous researches have been conducted using the combination of KSOM with ACO for clustering. For example, Mora et al. [22] proposed an ant-based method that takes benefit of the cooperative self-organizing ACO named as KohonAnts. It is designed as a clustering algorithm that is capable of grouping a set of the input samples into clusters with similar features same as KSOM behavior. The data is grouped without considering the class of the input pattern during the process. While Yang et al. [23] also applied ACO in their Ant-based of Self-Organizing feature Maps algorithm (ABSOM). This algorithm utilized the pheromone mechanism of ant colony system to memorize the history of the best matching unit and also adopted the state of transition rules of exploitation and exploration in ACO to determine the best matching unit. Chen et al. [27] proposed a new clustering method named AMC algorithm that can be accelerated by the use of a global memory bank, increase the radius of perception and also a density-based method that permits each ant to look for objects. This algorithm has reduced the times of region inquiry, hence, saved the clustering time.

Even though most of these hybrid algorithms are capable of grouping the data samples accurately into required classes, unfortunately the data samples are scatteredly mapped on the topology map. Moreover, it is hard to identify the separation boundary among the classes. Therefore, this paper has proposed a Pheromone-Based Self-Organizing Map (PKSOM) algorithm to refine and improve the scatteredness of the data in the clusters by modifying the KSOM algorithm using the pheromone concept from the Ant Clustering Algorithm (ACA). The ACA is chosen because of its strength, where it is robust, flexible, self-organize, good convergence and parallel [24]. It is a probabilistic technique for solving computation problem, which can be reduced to finding a good path through graph [25]. Besides, it can cluster the data samples into numbers of clusters and the total number of cluster is generated automatically [26].

This paper has been organized as follows. After presenting all concepts used in our method, in section Data Clustering, then we will discuss the proposed methods thoroughly, followed by a discussion on experimental works and analysis of results. Finally, we will conclude out description in the last section with a discussion of the obtained results and future works.

II. Data Clustering

A. Kohonen Self-Organizing Map (KSOM)

Developed by Teuvo Kohonen in year 1982, Kohonen Self-Organizing Map (KSOM) is an example of unsupervised training method for neural networks that implements the vector quantization [6]. Differing from the traditional vector quantization where KSOM task is to define how the mapping, m is ordered and how the input x is distributed on the map. In KSOM, there are a few factors that might influence the clustering results such as learning parameters, topology map and map sizes. The major steps in KSOM are in the distance calculation and the weights update. A KSOM unit computes the Euclidean Distance between an input x and its weight vector w. In the Kohonen one-dimensional network, the neighborhood of radius 1 of a unit at the k^{th} position consists of the units at the positions $k-1$ and $k+1$. The KSOM network and its algorithm are shown in Fig. 1 and Fig. 2.

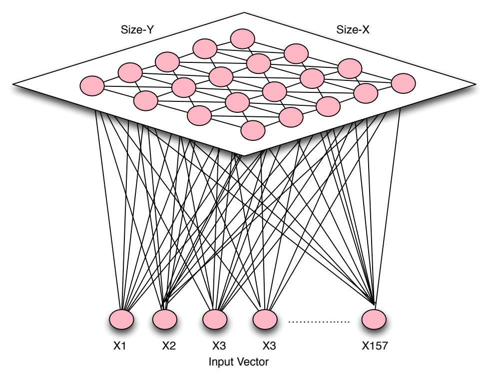


Figure 1. The KSOM network.

The performance of KSOM algorithm is measured using two measurements that commonly used in evaluating and measuring the self-organization algorithm: topological error and quantization error [10]. The topological error is used to measure the proportion of all data vectors for which first and second-best matching unit or winning unit are not adjacent vectors. If the value of topological error is lower then the KSOM preserves the topology is better. The quantization error is used to measure the average distance between each data vector and its best matching unit or winning unit. If the value of quantization error is small then it shows that the input vector is closer to its prototype.

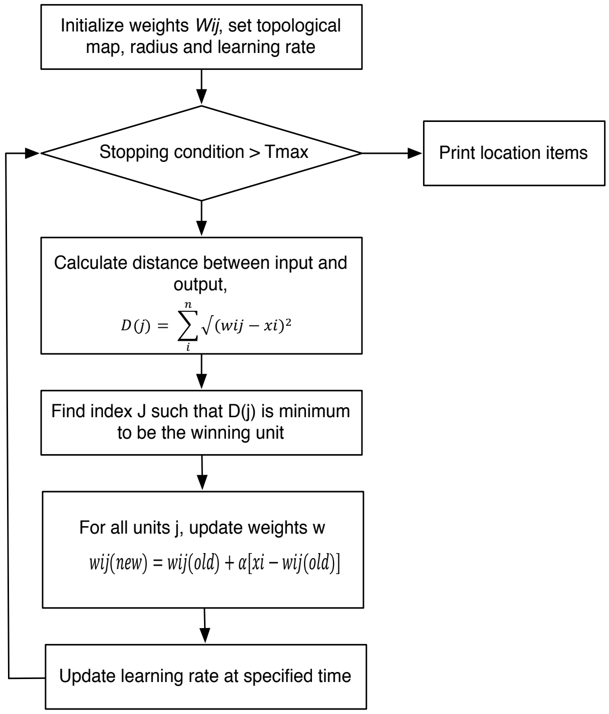


Figure 2. The KSOM Algorithm

B. *Ant Clustering Algorithm (ACA)*

The Ant Clustering Algorithm (ACA) [16] is an ant colony optimization algorithm that is designed for clustering purposes. It is a self-organizing algorithm where, the positive and negative feedback can display sudden modification that affect the pattern at the global level and the interaction is based on the cues and signals. Artificial ants are needed and the movements of these ants are based on the probabilities to pick up and drop down the object that are inversely proportional to the number of objects that are has experiences within a short period. Therefore, the ant will be more likely to deposit the object near larger clusters of objects. This algorithm applied the similarity idea where the degree of pair of data objects can reveal their probability of grouping into the same cluster. Fig. 3 shows the algorithm for ACA by Lumer et al. [11].

The ant can measure the similarity of the objects perceived within a local region and also identify the objects at the central site that is equally likely to group with the other objects. This algorithm used two types of pheromone for its searching strategy: cluster pheromone and object pheromone. Normally the cluster pheromone is used to lead or guide the ant to search for compact clusters while the object pheromone is to guide the ant to search for an object to be picked up and dropped. Both picking and dropping decisions require the evaluation of *f(i)*, which provides information on the similarity and density of the data items in the ant's local neighborhood.

```
procedure LUMER & FAEITA
Randomly scatter data items on toroidal grid
Randomly place agents on the toroidal grid
for t=1 to max iterations do
  j=random agent
  move agent j randomly by step size grid cells
  l= is agent j's grid position occupied by data item?
  e= is agent j's grid position occupied by a data item?
  if(l=TRUE) AND (e=FALSE) then
  i=data item carried by agent j
  drop=(random()<=pdrop(i)
  if drop = TRUE then
let agent j drop data item i at its current post
  end if
end if
if (l=FALSE) and (e=TRUE) then
  i=data item at agent j's grid position
  pick=random()<=Ppick(i)
  if pick=TRUE then
  let agent j pick up data item i
  end if
end if
end for
end procedure
```

Figure 3. The Ant Clustering Algorithm (ACA) algorithm

III. Pheromone-Based Kohonen Self-Organizing Map (PKSOM)

The proposed algorithm; Pheromone-Based Kohonen Self-Organizing Map (PKSOM) is a modified KSOM using a pheromone concept adapted from the ACA algorithm. As mentioned before, there are two main steps in KSOM that determine the self-organizing process: (1) distance calculation and (2) weight update. Some modification has been made to these two steps using the pheromone concept. The full PKSOM algorithm is shown in Fig. 4.

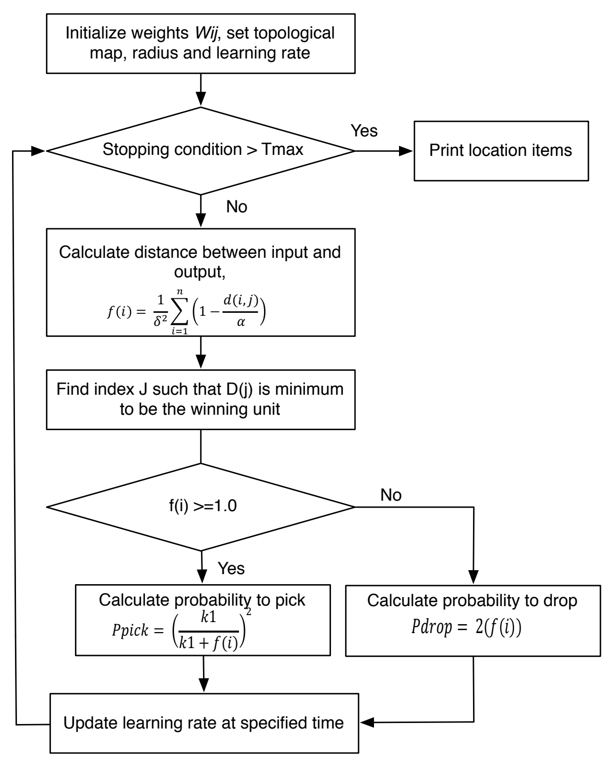


Figure 4. The PKSOM Algorithm

A. Distance Calculation

In KSOM, the distance calculation uses Euclidean Distance (ED); a distance between two points on a plane. The calculation is shown in (1) [1].

$$d(j) = \sum_{i=1}^{n} \sqrt{(wij - xi)^2} \quad (1)$$

The w_{ij} is representing the weight connection between input and output node, while x_i is the value of input *X*. This equation compares two objects across a range of variables and determines how dissimilar the objects are. In KSOM, the minimum value of ED is selected to be a winning unit. The minimum value also shows the two objects are very similar to each other.

For modification, the ED is replaced using Pheromone Density Measure (PDM) in ACA, as shown in (2). The pheromone density function is a way to measure the average of the similarity of objects o_i and other objects present in the neighbor δ, instead of looking at the distance, individually [11].

$$f(i) = \frac{1}{\delta^2} \sum_{i=1}^{n} \left(1 - \frac{d(i,j)}{\alpha}\right) \quad (2)$$

Here, the *d(i,j)* is to define the distance or dissimilarity between objects in the space of objects' attributes. While the α is a discriminant factor that defines the scale of dissimilarity and it is important for it to determine when two items should be or should not be located next to each other. The value selection for α is also crucial where this might affect the formation of clusters. If the α value is too large, there might not be enough discrimination between different items that may lead to the formation of clusters composed of item, which should not belong to the same cluster. On the other hand, if the value is too small, the distances between items in the space are amplified to the point where the items, which are relatively close in attribute space cannot be clustered together because the discrimination is too high.

B. Weights Update

Despite the pheromone density calculation, the other modified step is the weights' update. In original KSOM algorithm, the weight changes are calculated using Gaussian function [2], as shown in (3). The Gaussian function is used as a decreasing function of the grid distance between objects. Due to the collective learning scheme in KSOM, the input signals, which are near to each other, will be mapped on neighboring neurons. Thus, the topology inherently present in the input signals will be preserved in the mapping. The *rk* and *rc* represent the two objects to be calculated, while the δ is representing the neighborhood radius.

$$\Delta w = exp\left(-\frac{(rk - rc)^2}{2\delta^2}\right) \quad (3)$$

In order to ensure this algorithm works well with pheromone density calculation, the Gaussian function is replaced with the probability to pick up (4) and probability to drop (5) the object in ACA. Deneubourg et al. [17] have proposed these probabilities based on the corpse clustering and the larvae sorting in ants; where the isolated item should be picked up and dropped at some other location where more items of that type are present. The decisions to drop and pick the object are random and influenced by the data items in the neighborhood. The probability of dropping an item might increase if the surrounded neighborhood data is similar. In contrast, the probability of picking an item might increase if the surrounded neighborhood is dissimilar.

$$Ppick = \left(\frac{k1}{k1 + f(i)}\right)^2 \quad (4)$$

$$Pdrop = \begin{cases} 2(f(i) \; when \; f(i) < k2 \\ 1 \; when \; f(i) \geq k2 \end{cases} \quad (5)$$

For both equations, the *f(i)* is the PDM, while the k_1 and k_2 are threshold constants. The selected value for both *k* is 1 and this is according to Lumer et al. [11], where they have defined a distance or dissimilarity between objects in the space of object attributes:

- If two objects are similar or identical, then the $d(o_i,o_j) = 0$
- If two objects are not similar or identical, then the $d(o_i,o_j) = 1$

IV. EXPERIMENTAL WORKS AND ANALYSIS OF RESULTS

A. Experimentals Works

The proposed algorithm has been tested using three different datasets; (1) Iris, (2) Glass and (3) Wood datasets (as shown in Table 1). The Iris dataset consists of 150 samples, which represent 3 species; sentosa, virginica and versicolor. While for Glass's data, the dataset has 216 samples and can be categorized into 2 categories; window and non-window glass. Moreover, the dataset for Wood that owned 5040 samples of data comprises 52 tropical Wood species and this dataset can also be categorized according to the most dominant features; the pore size. There are three sizes of pores; small, medium-sized and large and the list of tropical wood species based on pores sizes is shown in Table 2.

TABLE 1: THE DATASET USED TO EVALUATE THE ALGORITHM

Datasets	No of Samples	Attributes	Category
Iris	150	4	3
Glass	215	9	2
Wood	5040	157	3

TABLE 2. TROPICAL WOOD SPECIES BASED ON PORES SIZES

Pore Sizes	Wood Species
Small	mataulat
Medium-Sized	balau, bintangor, bitis, chengal, gerutu, giam jelutong, kapur, kasai, kekatong, keledang, keranji, kulim, machang, medang, melunak, perupok, redbalau
Large	bintangor, durian, gerutu2, kapur, kasai, keledang, keruing, machang, merantibakau, redbalau, rubberwood, sesendok

The training processes are done repetitively and each dataset used different parameters; such as learning rate and number of epochs. These values have been personalized to each dataset and parameters values are obtained from the KSOM training process for performance comparison purposes. Table 3 shows the full parameters used for these three datasets. The selection of topological map for all datasets is based on Alhoniemi et al. [18], calculated using:

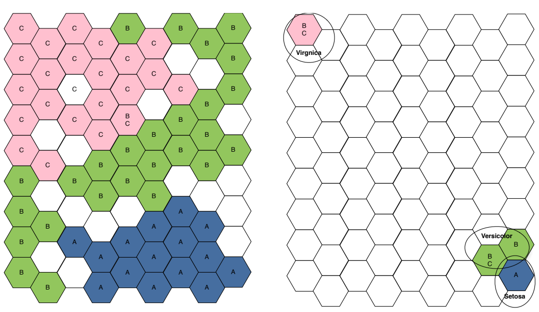
$$mapsize = 5(\,no\ of\ sample^{\,0.54321}) \qquad (6)$$

TABLE 3. FULL PARAMETERS USED FOR ALL DATASETS

Parameters	Datasets		
	Iris	Glass	Wood
Topological Map	9x9 (81 nodes)	9x9 (81 nodes)	23x23 (529 nodes)
Sample number	150	216	5040
Categories/ Number of Clusters to be clustered	3 (Setosa, Versicolor & Virginica)	2 (Window-Glass & Non-window Glass)	3 (Small, Medium-sized & Large pores)

B. Results

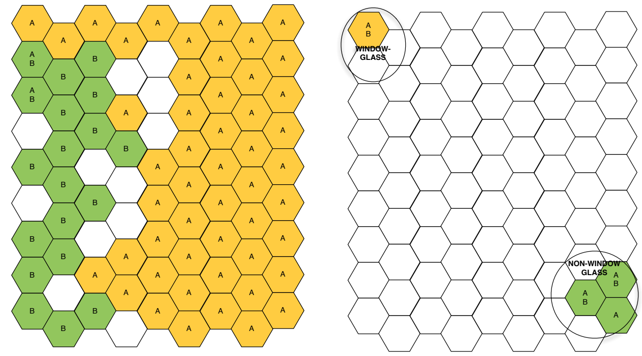
As a result, it is obviously seen that the PKSOM has improved and refined the scatteredness of clustering data by increasing the density of the data in clusters. Most of the data are well clustered and closed to each in the same cluster even though there is a few dislocated data occurred and performed the overlapped clusters. Fig. 5 shows the results for Iris dataset using both algorithms. KSOM has clustered the Iris dataset into three big clusters (consists of a few clusters for every species) according to the species. However, the PKSOM has also clustered the dataset into three different clusters with high density of data in each cluster.



(a) (b)

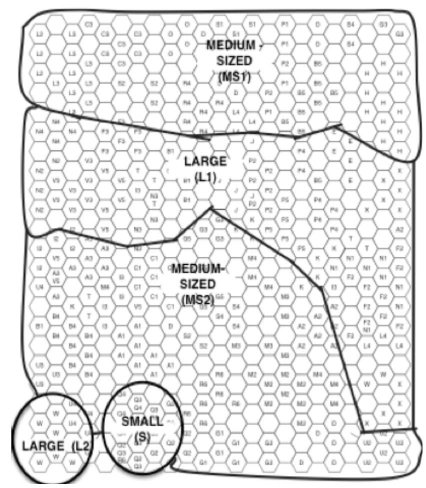
Figure 5. Clustering Results using (a) KSOM and (b) PKSOM for Iris Dataset

Fig. 6 shows the clustering results performed by KSOM and PKSOM for Glass dataset, where the KSOM has separated the dataset into two categories: the window glass (on the upper part of the map) and non-window glass (on the lower part of the map). Conversely, the PKSOM has clearly separated the dataset into two different classes with a minimum number of cluster nodes and high density of data in each node. While for Wood dataset, KSOM has clustered the whole dataset into five big clusters: one cluster for small pores, two clusters for medium-sized pores and two clusters for large pores. However, the PKSOM has clustered the Wood dataset into three desired clusters, accurately. This is shown in Fig. 7.

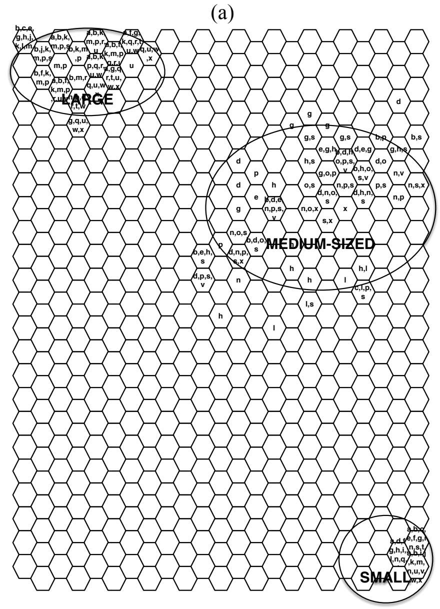


(a) (b)

Figure 6. Clustering Results using (a) KSOM and (b) PKSOM for Glass Dataset



(a)



(b)

Figure 7. Clustering Results using (a) KSOM and (b) PKSOM for Wood Dataset

 ISBN: 978-1-61208-352-0

TABLE 4: FULL RESULT FOR ALL DATASET

Dataset	Map Size	Cluster Usage (Cluster Node With data)		Average Cluster Density of single node	
		KSOM	PKSOM	KSOM	PKSOM
Iris	9 x 9 (81 cluster nodes)	64	4	±2	±38
Glass	9 x 9 (81 cluster nodes)	69	4	±3	54
Wood	23 x 23 (529 cluster nodes)	399	71	±13	±71

Conclusively, as shown in Table 4, the KSOM has clustered Iris dataset into 64 clusters, Glass dataset into 69 clusters and wood dataset into 399 clusters. The percentage of cluster usage (the cluster node that consists of data) for Iris dataset is 79.01%, Glass dataset is 85.19% and for wood dataset is 75.43%. While for proposed algorithm, PKSOM, the percentage of cluster usage for every dataset are 4.94% (Iris dataset), 4.94% (Glass dataset) and 23.92% (wood dataset), as shown in Fig. 8.

Even though the KSOM's average of cluster usage is higher than the PKSOM, but the average cluster density of each cluster is lower compared to PKSOM (shown in Fig. 9). The PKSOM has the produced high average cluster density for every dataset; for Iris, the average cluster density is 38, Glass dataset average cluster density is 54 and for wood dataset is 70.99. However, the average cluster densities for every dataset produced by KSOM are very low; 2.34 (Iris dataset), 3.13 (Glass dataset) and 12.63 (Wood dataset).

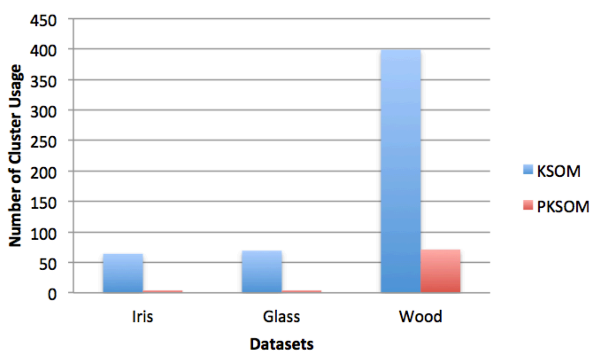


Figure 8. The Difference of Cluster Usage between KSOM and PKSOM

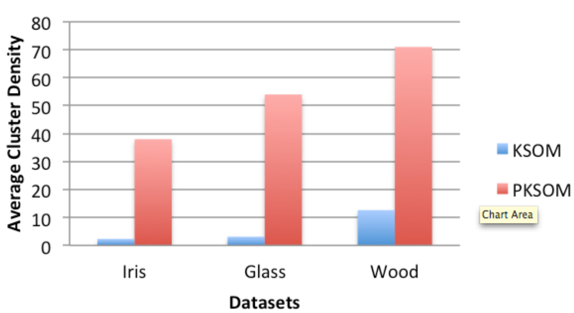


Figure 9. The Average Cluster Density of single node

For performance evaluation, we have compared the PKSOM result with previous methods; KohonAnts [22] and ABSOM [23] based on Iris results. Fig. 10 shows the results for KohonAnts, ABSOM and PKSOM. Both methods: KohonAnts and ABSOM have produced good classification result but the clustered data is scatteredly mapped into a two dimensional map. However, it is clearly seen that PKSOM has clustered the Iris dataset into three clusters, same as KohonAnts and ABSOM but with a minimum number of cluster usage. This is proved that PKSOM has refined the scatteredness of the data, thus improved the separable boundary between clusters.

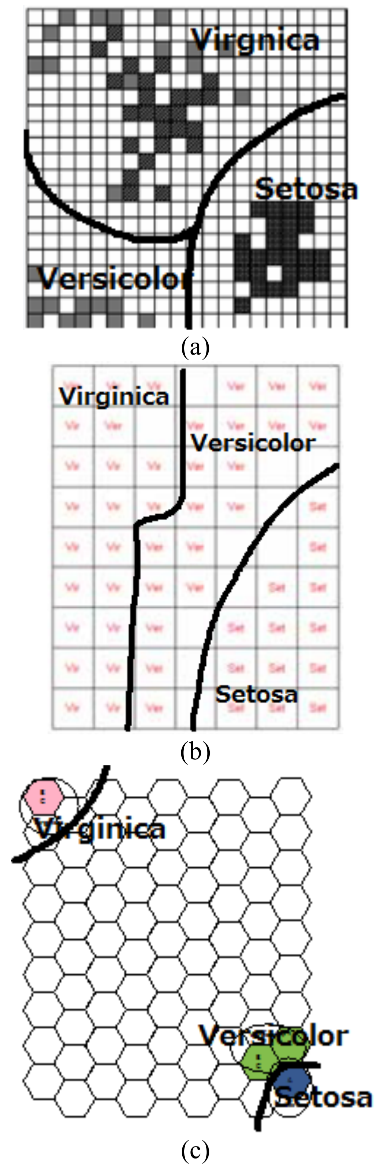


Figure 10. Clustering Results using (a) KohonAnts, (b) ABSOM and (c) PKSOM for Iris Dataset

V. CONCLUSION AND FUTURE WORK

In conclusion, it is obviously seen and proven that the PKSOM has improved and refined the scatteredness of clustering data by increasing the density of the data in clusters. Most of the data for all datasets are well clustered and closed to each other in the same cluster. However, there are a few

 ISBN: 978-1-61208-352-0

dislocated data that performed the overlapped clusters. These overlapped clusters consist of at least two different species or category in every overlapped cluster. According to the result, also, we can conclude that PKSOM also can deal with high dimensional dataset, such as wood dataset. Furthermore, we will perform further test using different high dimensional and sparse dataset to investigate the effectiveness of the proposed method. In addition, we will make some refinement to the algorithm in order to solve overlapped clusters problem and increase the classification accuracy.

ACKNOWLEDGEMENT

The authors would like to thank Ministry of Education Malaysia for funding this research project through a Research University Grant of Universiti Teknologi Malaysia (UTM), project titled "Dimension Reduction & Data Clustering for High Dimensional & Large Dataset" (04H40). Also, thanks to the Research Management Center (RMC) of UTM for providing an excellent research environment in which to complete this work

REFERENCES

[1] R. Rojas, “Unsupervised Learning and Clustering Algorithms,” in *Neural Networks*, Berlin: Springer Berlin Heidelberg, 1996, pp. 99–121.

[2] T.Kohonen, *Self-Organizing Map.* Springer Series in Information Sciences 3rd Edition, Springer, 2000.

[3] J. L. Giraudel and S. Lek, “A comparison of self-organizing map algorithm and some conventional statistical methods for ecological community ordination,” *Ecol. Modell.*, vol. 146, no. 1–3, Dec. 2001, pp. 329–339.

[4] A. Dekker, “Kohonen neural networks for optimal colour quantization,” *Network-computation Neural Syst.*, vol. 5, no. 3, 1994, pp. 351–367.

[5] V. Emamian, A. H. Tewfik, E. Shizcorningcom, and L. J. Jacobs, “Robust Clustering of Acoustic Emission Signals Using Neural Networks and Signal Subspace Projections” , 2003, pp. 276–286.

[6] T. Honkela, S. Kaski, K. Lagus, T. Kohonen, and N. Networks, “Exploration of Full-Text Databases with Self-organizing Maps,” in *IEEE International Conference on Neural Networks*, 1996, pp. 56 – 61.

[7] W. Wan and D. Fraser, “M2dSOMAP: Clustering and Classification of Remotely Sensed Imagery by Combining Multiple Kohonen Self-organizing Maps and Associative Memory,” no. 41, 1993, pp. 2464–2467.

[8] P. X. P. Xu, C.-H. C. C.-H. Chang, and A. Paplinski, “Self-organizing topological tree for online vector quantization and data clustering.,” *IEEE Trans. Syst. man Cybern. Part B Cybern. a Publ. IEEE Syst. Man Cybern. Soc.*, vol. 35, no. 3, 2005, pp. 515–526.

[9] F. Forkan and S. M. Shamsuddin, “Kohonen-Swarm Algorithm for Unstructured Data in Surface Reconstruction,” *2008 Fifth Int. Conf. Comput. Graph. Imaging Vis.* , Aug. 2008, pp. 5–11.

[10] Z. M. Zin, M. Khalid, E. Mesbahi, and R. Yusof, “Data Clustering and Topology Preservation Using 3D Visualization of Self Organizing Maps,” vol. II, 2012, pp. 1–6.

[11] E. D. Lumer and B. Faieta, “Diversity and Adaptation in Populations of Clustering Ants,” in *The Third International Conference on Simulation of Adaptive Behavior: From a Animals to Animats 3*, 1994, pp. 501–508.

[12] P. Vaijayanthi, A. M. Natarajan, and R. Murugadoss, “Ants for Document Clustering,” vol. 9, no. 2, pp. 493–499, 2012.

[13] C. Tsang and S. Kwong, “Ant Colony Clustering and Feature Extraction for Anomaly Intrusion Detection,” in *Swarm Intelligence in Data Mining*, vol. 34, Springer Berlin Heidelberg, 2006, pp. 101–123.

[14] J. Handl, J. Knowles, and M. Dorigo, “Ant-based clustering and topographic mapping.,” *Artificial Life*, vol. 12, no. 1, Jan. 2006, pp. 35–62.

[15] S. Bala, S. I. Ahson, and R.P.Agarwal, “A Pheromone Based Model for Ant Based Clustering,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 11, 2012, pp. 180–183.

[16] O. A. M. Jafar and R. Sivakumar, “Ant-based Clustering Algorithms: A Brief Survey,” *Int. J. Comput. Theory Eng.*, vol. 2, no. 5, 2010, pp. 787–796.

[17] J. L. Deneubourg, S.Goss, N.Franks, A.Sendova-Franks, C.Detrain, and L. Chretien, “The Dynamics of Collective Sorting Robot-Like Ants and Ant-Like Robots,” in *Proceeding of the First International Conference on Simulation of Adaptive Behavior on From Animals to Animats*, 1990, pp. 356–363.

[18] E. Alhoniemi et.al. (April,2014), *Laboratory of Computer and Information Science*, 2005. [Online]. Available: http://www.cis.hut.fi/projects/somtoolbox/documentation/.

[19] M. Mishra, “Kohonen Self Organizing Map with Modified K-means clustering For High Dimensional Data Set,” vol. 2, no. 3, 2012, pp. 34–39.

[20] E. Mohebi and M. N. M. Sap, “An Optimized Hybrid Kohonen Neural Network for Ambiguity Detection in Cluster Analysis Using Simulated Annealing,” in *Enterprise Information Systems*, Springer Berlin Heidelberg, 2009, pp. 389–401.

[21] A. H. Abdullah, “An Optimized Clustering Algorithm Using Genetic Algorithm and Rough set Theory based on Kohonen self organizing map,” *Int. J. Comput. Sci. Inf. Secur.*, vol. 8, no. 4, 2010, pp. 39–44.

[22] A. M. Mora, C. M. Fernandes, J. J. Merelo, V. Ramos, J. L. J. Laredo, and A. C. Rosa, “KohonAnts: A Self-Organizing Ant Algorithm for Clustering and Pattern Classification,” *Artificial Life XI*, 2008, pp. 428–435.

[23] C. C. Yang and S. Chi, “An Ant-Based Self-Orgazining Feature Maps Algorithm,” in *5th Workshop On Self-Organizing Maps*, 1997, pp. 65–74.

[24] W. Dai, S. Liu, and S. Liang, “An Improved Ant Colony Optimization Cluster Algorithm Based on Swarm Intelligence,” *J. Softw.*, vol. 4, no. 4, 2009, pp. 299–306.

[25] I. Michelakos, N. Mallios, E. Papageorgiou, and M. Vassilakopoulos, “Ant Colony Optimization and Data Mining,” in *Next Generation Data Technologies for Collective Computational Intelligence*, Springer Berlin Heidelberg, 2011, pp. 31–60.

[26] Q. Chen and J. Mo, “Optimizing the Ant Clustering Model Based on K-Means Algorithm,” *2009 WRI World Congr. Comput. Sci. Inf. Eng.*, no. 4, 2009, pp. 699–702.

[27] J. Chen, J. Sun, and Y. Chen, “A New Ant-based Clustering Algorithm on High Dimensional Data Space,” in *Complex Systems Concurrent Engineering*, Springer London, 2007, pp. 605–611.

Collision Estimation Using Single Camera

Discussion under the condition of constant velocity

Ryunosuke Ikeno[1], Kazuyuki Ito[2]
Dept. of Electrical and Electronics Engineering
Hosei University
Tokyo, Japan
e-mail: ryunosuke.0416@gmail.com, ito@hosei.ac.jp

***Abstract*— Robots and autonomous vehicles that operate in complex dynamical environments have attracted considerable attention in recent years. Avoiding obstacles and finding a passable route to the destination is one of the important and basic functions of mobile robots. In this paper, we take our cue from the obstacle-avoidance mechanism in animals, and discuss how to enable robots to find a passable route using only two-dimensional images from a single usual camera without any distance information. We simulate the dynamic changes in visual information and obtain the necessary conditions for a mobile robot to determine impending collisions with an obstacle in its path.**

***Keywords*- Autonomous vehicles; Estimation of collision; visual information single camera.**

I. INTRODUCTION

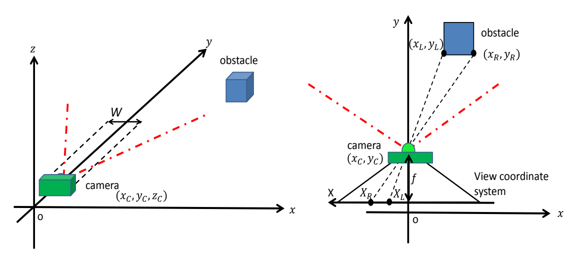
In recent years, robots and autonomous vehicles that operate in complex dynamical environments such as daily life have attracted considerable attention. Autonomous navigation of automobiles or personal robots is among these fertile areas of research. Conventional robots were designed for well-known, simple environments, such as a factory. Thus, conventional methodology does not work effectively in designing robots that operate in unknown dynamic environments.

Considerable research has been conducted to solve this problem and develop effective autonomous robots. DARPA ground challenge [1] and Google car [2] are one of them. In these conventional studies, the most common approach to such a problem is to create three-dimensional models of the environment in question. In this approach, robots have sensors, such as a laser range-finder, to measure distances to obstacles and create a precise three-dimensional model of the environment [3]-[7]. A passable route is obtained by calculations that use information from the three-dimensional model. While this approach is effective, the mechanism to obtain the passable route is very different from that of animals. Even lower-level animals and insects can act quickly to avoid obstacles, in spite of their small brains. They have no distance sensor and their brain is too small to find a passable route to their destination as quickly as they do. How these animals are nonetheless able to do so is still an open question. However, research in ecological psychology has revealed that animals can evaluate the time-to-contact for an object in their path based solely on available visual information and without any distance information. Information about the time-to-contact is perceived directly and no complex computation is presumably required.

In this paper, we focus on this mechanism in animals and discuss how to enable robots to find a passable route to their destination using two-dimensional images from a single camera without any distance information. We simulate the dynamic changes in visual information and establish the necessary conditions to accurately determine impending collisions with an obstacle.

II. MODEL AND DEFINITION

We define the global coordinate system and the view coordinate system as shown in Fig. 1 and Fig. 2, respectively. Table 1 elaborates on the meaning of the symbols used in these figures. The robot (camera) moves on the y-axis at a constant velocity and the direction of the camera is fixed, as shown in Fig 1(b). There is a static obstacle in the robot's path, and the position of the obstacle in the global coordinate system is converted to that in the view coordinate system by (1) and (2).



(a) Side view (b) Top view

Figure 1. Global coordinate system.

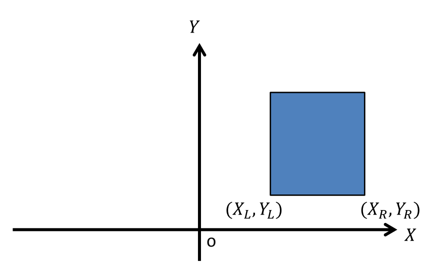


Figure 2. View coordinate system.

 ISBN: 978-1-61208-352-0

TABLE I. DEFINITION

f	Focal length
W	Width of the body
$(X_{L(R)}, Y_{L(R)})$	Position of the obstacle in the view coordinate system
$(x_{L(R)}, y_{L(R)})$	Position of the obstacle in the global coordinate system
(x_C, y_C)	Position of the camera in the global coordinate system

$$X_L = f \times \frac{x_L}{y_L - y_C} \quad (1)$$

$$X_R = f \times \frac{x_R}{y_R - y_C} \quad (2)$$

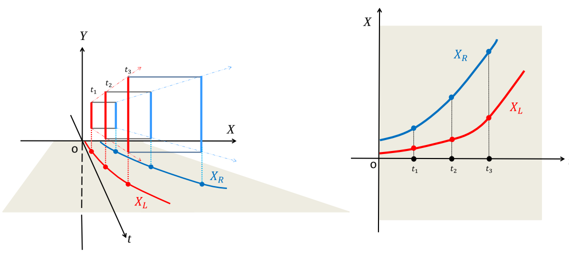


Figure 3. Temporal response of the edges of object.

Fig. 3 illustrate the temporal response of the edges of the object in the view coordinate system. In this study, we focus on this temporal change, and discuss how to determine collision to the obstacle.

III. SIMULATION

A. Case of ignoring body size

In this subsection, we assume that the body of the robot is a point and has no width. We conduct simulations for two cases, as shown in Fig. 4 and Fig. 5. Fig. 4 shows the case where the robot comes into contact with the obstacle and Fig. 5 represents one where the robot successfully evades it. In both cases, the robot moves in a straight line at the same constant velocity. By comparing these results, we can learn how to estimate future crashes using the visual information at hand.

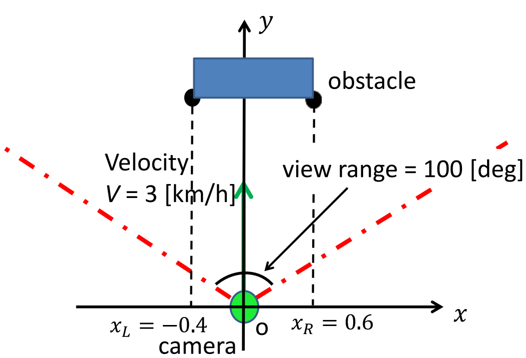


Figure 4. Case A-1: Robot contacts the obstacle.

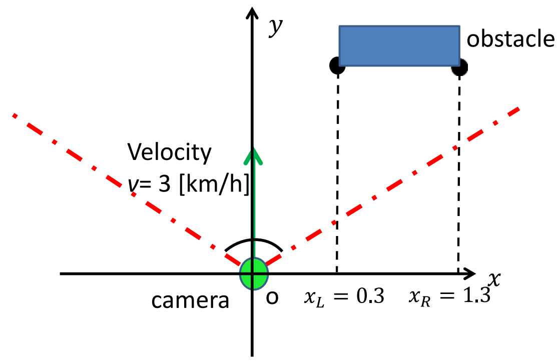


Figure 5. Case A-2: Robot does not contact the obstacle.

Fig. 6 and Fig. 7 show the results of our simulation. These graphs show the change in position of the obstacle relative to the robot's visual coordinate system as it moves forward. From these results, it is obvious that if both sides of the obstacle appear to expand outwards from the robot's visual perspective, the robot will hit the obstacle. On the other hand, if both sides of the obstacle appear to move in one direction from the perspective of the robot's visual coordinate system, the robot can evade the obstacle.

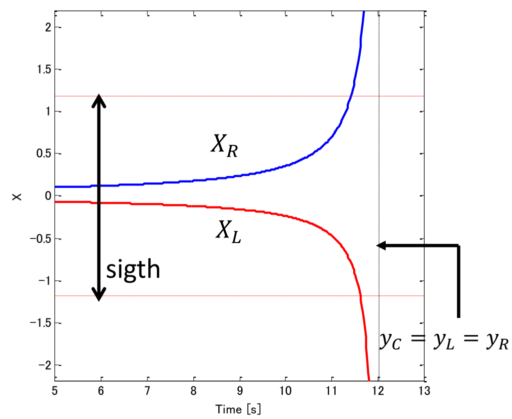


Figure 6. Result of Case A-1

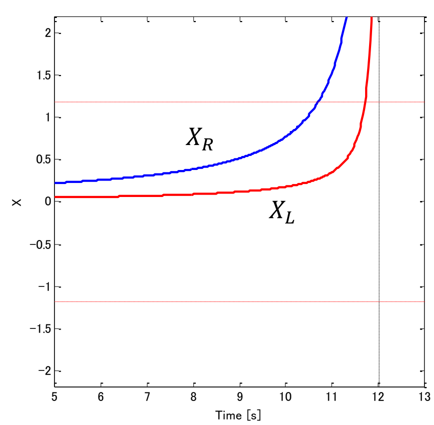


Figure 7. Result of Case A-2

B. Case of considering body size

In this subsection, we assume that the body of the robot has a certain non-negligible width. We conduct simulations with the robot at different distances from the obstacle, as shown in Fig. 8 and Table 2. In all cases, the robot moves in a straight line at the same constant velocity.

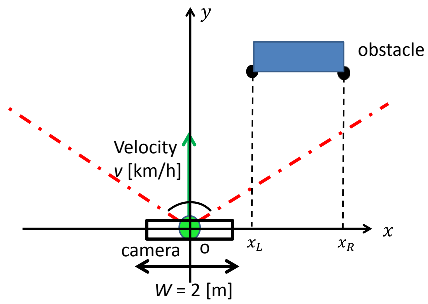


Figure 8. Case B: Robot has a certain width

TABLE II. POSITION OF THE OBSTACLE

Case B-1	$x_L = 0.1, x_R = 1.1$
Case B-2	$x_L = 0.5, x_R = 1.5$
Case B-3	$x_L = 1.0, x_R = 2.0$
Case B-4	$x_L = 1.5, x_R = 2.5$
Case B-5	$x_L = 2.0, x_R = 3.0$

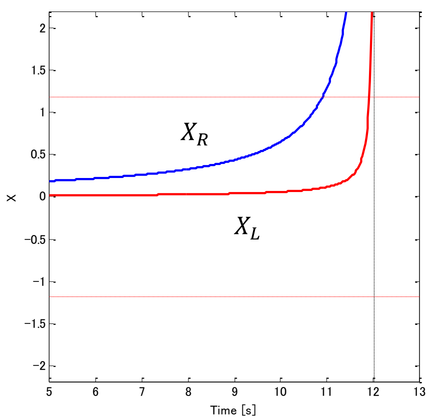


Figure 9. Result of Case B-1

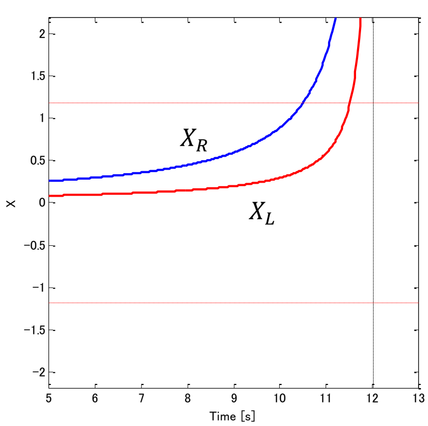


Figure 10. Result of Case B-2

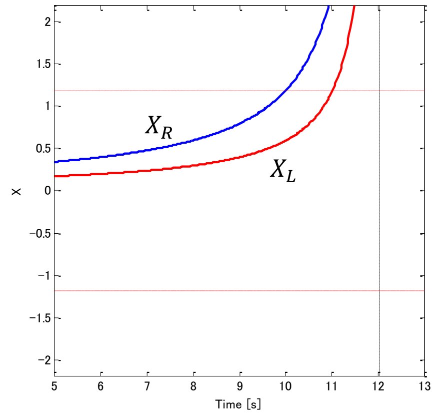


Figure 11. Result of Case B-3

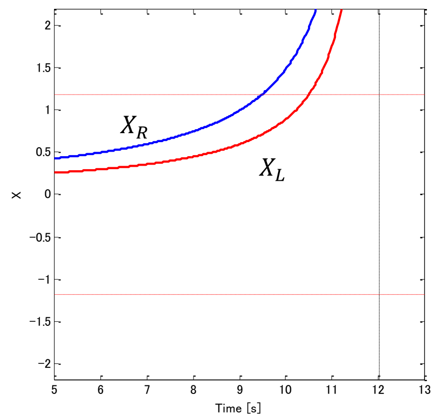


Figure 12. Result of Case B-4

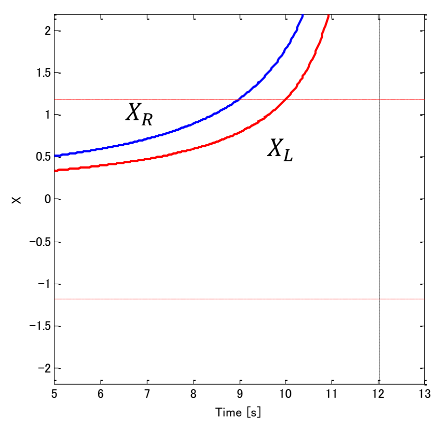


Figure 13. Result of Case B-5

Figs. 9 to 13 show the simulation results. In all cases, both sides of the obstacle appear to move in the same direction from the robot's visual perspective. Thus, according to the condition stated in Section III.A, the robot should pass the obstacle by. However, in this simulation, the width of the robot's body is 2m, because of which it hits the obstacle in cases B-3, B-4 and B-5. From these results, we can conclude that the condition in Section III.A is not sufficient for the robot to avoid the obstacle when it is sufficiently large in size.

To take the size of the robot's body into consideration, we focus on the temporal changes in values of X_L and X_R, the horizontal coordinates that demarcate the width of the obstacle, with the robot's motion. Fig. 14 shows the temporal change in the value of X_L. We see that this change depends on the distance between the robot and the obstacle. Therefore, we can calculate if a collision will occur from the change in the horizontal coordinates of the obstacle relative to the robot's motion, if the width of the robot's body and its velocity are known.

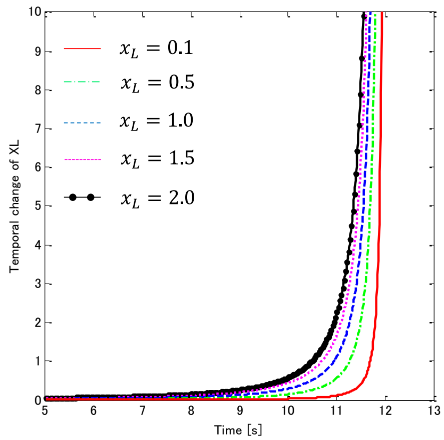


Figure 14. Temporal change of X_L for different settings of x_L

C. *Case of different velocities*

In this subsection, we consider the influence of the velocity of the robot on obstacle avoidance. The simulation setting is the same as in Section III.B, except for the velocity of the robot.

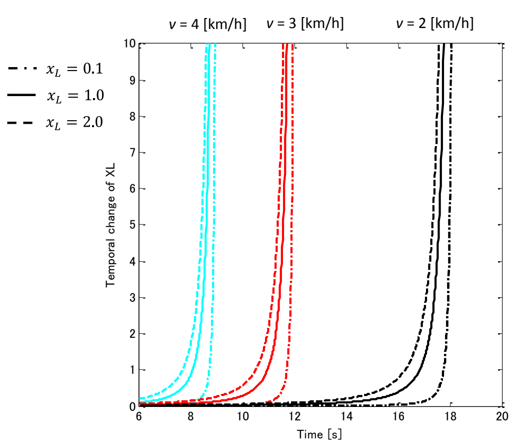


Figure 15. Temporal changes of X_L for different velocities

Fig. 15 shows the results. We see that the temporal change in X_L depends on the robot's velocity in addition to its distance from the obstacle. Thus, to determine if a collision will occur, the velocity of the robot relative to the obstacle is also required.

Animate living beings do not have velocity sensors. They estimate relative velocity using visual information. Thus, the determination that a collision will occur or that a passable route that avoids obstacles is available can only be made through visual information.

Our future research in this area will focus on the calculation of relative velocity using available visual information.

IV. CONCLUSION

In this paper, we discuss a method for a mobile robot to detect impending collision with an obstacle in its path and to find a passable route using only two-dimensional images from a single usual camera without any distance information. We simulated the dynamic changes in visual information and determined the necessary conditions to determine future collisions with an obstacle.

Our future research will be geared towards estimating the relative velocity of a mobile robot with respect to obstacles using visual information, and applying it to determine impending collisions.

ACKNOWLEDGEMENT

This research was partially supported by the Japan Society for the Promotion of Science through the Grant-in-Aid for Scientific Research (C) 24500181.

REFERENCES

[1] U. Ozguner, C. Stiller, and K. Redmill, "Systems for safety and autonomous behavior in cars," The DARPA Grand Challenge experience. Proceedings of the IEEE, 2007, vol. 95, 2, pp. 397-412.

[2] "How Google's Self-Driving Car Works – IEEE Spectrum," Spectrum.ieee.org, Feb. 26. 2013.

[3] S.Thrun, Robotic Mapping: A Survey," CMU Technical Report, CMU-CS-02-111, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 2002.

[4] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," Proc. AAAI National Conference on Artificial Intelligence (Edmonton 2002), 2002, pp. 593-598.

[5] A. Diosi, and L. Kleeman, "Laser Scan Matching in Polar Coordinates with Application to SLAM," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005), 2005, pp. 3317–3322.

[6] M. Bosse, P. Newman, J. Leonard, and S. Teller, "Simultaneous Localization and Map Building in Large-Scale Cyclic Environments Using the Atlas Framework," The International Journal of Robotics Research, Dec. 2004, vol. 23, 12, pp. 1113–1139.

[7] A. Nguyen, N. Martinelli, Tomatis, and R. Siegwart, "A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics," Int. Conf. Intelligent Robots and Systems, 2005, pp. 1929–1934.

Velocity Estimation from Visual Information using Environmental Property

Hitoki Takase, Kazuyuki Ito
Dept. of Electrical and Electronics Engineering
Hosei University
Tokyo, Japan
e-mail:tmrmeme7@gmail.com, ito@hosei.ac.jp

***Abstract*—In ecological psychology, it is considered that animals and insects use visual instead of distance information. In this paper, we take the mechanism of animals into consideration and address the method used to estimate the velocity of a robot by employing only one camera. Simulations are conducted and their accuracy is discussed.**

Keywords-velocity estimation; monocular camera; ecological psychology.

I. INTRODUCTION

Intelligent safety systems for automobiles, such as autonomous collision avoidance and navigation systems, have recently attracted considerable attention, leading to extensive research into developing effective autonomous vehicles. The DARPA ground challenge and the Google car are two examples [1]-[3]. The most common approach in these conventional studies is to create three-dimensional models of the environment. In this approach, the vehicles have sensors, such as laser range-finders, to measure distances to obstacles and create a precise three-dimensional model of the environment [4]-[8]. However, the creation of this environment model and the extraction of useful information involve huge computational costs. In contrast, although lower animals and insects do not have such distance sensors and their brains are very small, they behave adaptively even in unknown environments [9]-[11]. In ecological psychology, it is presumed that animals and insects use visual information instead of distance information.

This study considers the navigation mechanism of animals and addresses the method used to estimate the velocity of a robot using visual information. We focus on the ecological niche framework and assume that the average distance from obstacles is constant. Under this assumption, we derive an equation to estimate velocity from visual information. Simulations are conducted, and the error rate arising from the error of average distance is discussed.

This paper consists of the following parts. Section II defines the global coordinate system and explains the problem domain and the assumption in this paper. Section III describes a preliminary experiment that was conducted to discuss the validity of the assumption. Section IV proposes a method to estimate the velocity, and in Section V, simulations are conducted and the error rate of the proposed method is discussed. Section VI concludes this paper.

II. PROBLEM DOMAIN

In this study, we address the method of estimating the velocity of a robot by using only one camera; we employ no other sensors. In general, it is impossible to estimate velocity using a single camera. Hence, we make an assumption based on ecology. In ecology, it is reported that each animal occupies and exploits a unique niche. The size and variation of obstacles in its environment are aspects of this niche. Accordingly, we focus on the average distance from obstacles. The average distance depends on the niche and does not change rapidly. Therefore, in this study, we assume that the average distance from visible obstacles is known (i.e., it can be learned) and constant. Under this assumption, we derive an equation to estimate velocity from visual information. In practice, the average distance shows small variations; thus, we discuss the error rate of the estimated velocity arising from the error of average distance. The details are as follows.

We define the global and view coordinate systems as shown in Figure. 1, and we define the estimated average of distance as shown in Figure 2.

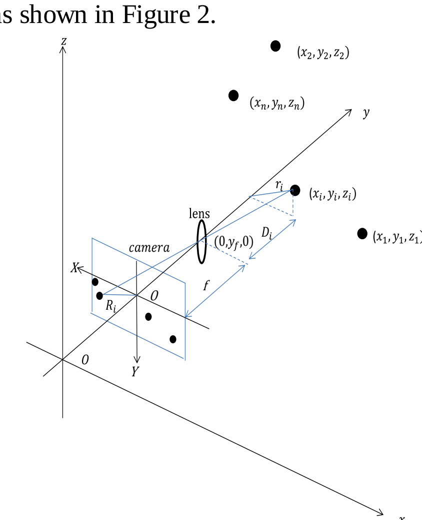


Figure 1. Definition of coordinate systems.

 ISBN: 978-1-61208-352-0

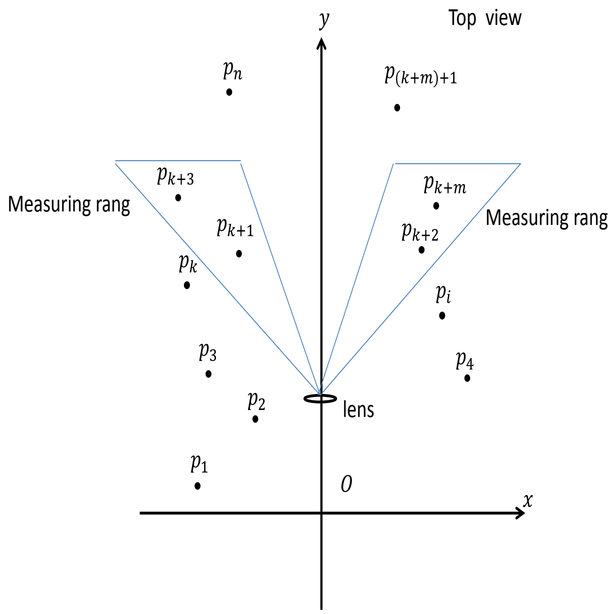


Figure 2. Definition of average distance.

The camera is located on the y-axis of the global coordinate system and moved forward at a constant velocity v. The lens is positioned at (0, y_f, 0). There are several static object, and the coordinates of their characteristic points p_i are denoted by (x_i, y_i, z_i), where n is the number of characteristic points. D_i is the distance between the lens and the i-th point, and it is expressed as $D_i = y_i - y_f$. R_i is the distance from the origin in the view coordinate system, r_i is the distance from the y-axis in the global coordinate system, and f is the focal length of the camera.As described above, we assume that each D_i is unknown. However, only the average distance from the visible characteristic points is estimated, as shown in (1):

$$\widehat{D} \approx \overline{D} = \frac{1}{m}\sum_{i=k+1}^{k+m} D_i \quad (1)$$

where only p_{k+1} to p_{k+m} are visible among all characteristic points (p_1 to p_n), as shown in Figure. 2. $\overline{D}$ is the actual average distance from the visible characteristic points, and $\widehat{D}$ is the predefined constant, estimated value of $\overline{D}$.

III. PRELIMINARY EXPERIMENT

To discuss the validity of the assumption written in Section II, we conducted preliminary experiment.

Figure. 3 shows setting of the experiment. We measured distance l_i and angle θ_i ever 5° from 30° to 75° and from 105° to 150°,then we move forward 2 m and measured again. From l_i and θ_i we calculated D_i and $\overline{D}$. Figure. 4 and Figure. 5 shows environments, and Table 1 and 2 shows experiment results

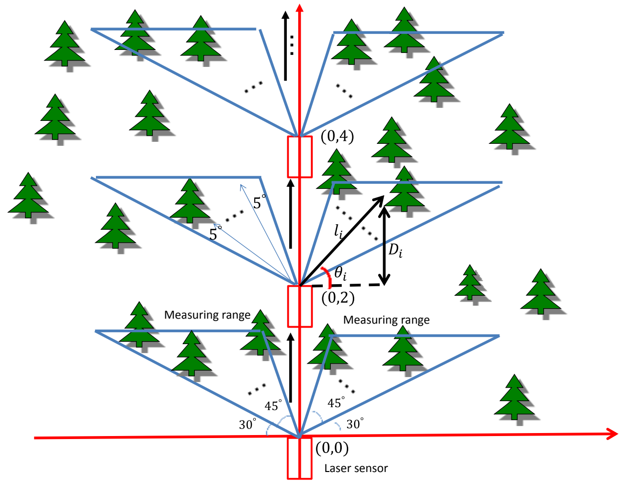


Figure 3. Range of measurement

Figure 4. Environment 1

Figure 5. Environment 2

 ISBN: 978-1-61208-352-0

TABLE 1. RESULT OF ENVIRONMENT 1

Measurement position	$\bar{D}$
(0,0)	2.73
(0,2)	2.99
(0,4)	2.87
(0,6)	2.96
(0,8)	2.88

TABLE 2. RESULT OF ENVIRONMENT 2

Measurement position	$\bar{D}$
(0,0)	12.15
(0,2)	12.03
(0,4)	12.83
(0,6)	12.32
(0,8)	12.13

Environment 1 is a passage way in a building and there is walls on its both sides. Though there are some objects, influence of these objects is filtered by the calculation of average. Thus the average distance does not change rapidly as shown in Table. 1. Environment 2 is a road between a building and a green zone, and the similar tendency is observed as shown in Table. 2. From these results, we can confirm that $\bar{D}$ is dependent on the environment, however, $\bar{D}$ does not change rapidly.

IV. VELOCITY ESTIMATION

Now, we estimate the velocity of the camera using the visual information R_i. From the geometric relationship, R_i is given by (2):

$$R_i = f\frac{r_i}{D_i} \tag{2}$$

Differentiating both sides of (2), we obtain (3):

$$\dot{R}_i = -f\frac{r_i}{{D_i}^2}\dot{D}_i \tag{3}$$

Dividing (2) by (3), we obtain (4):

$$\frac{R_i}{\dot{R}_i} = -\frac{\frac{fr_i}{D_i}}{\frac{fr_i}{{D_i}^2}\dot{D}_i} \tag{4}$$

From (4), we obtain (5):

$$D_i = -\frac{R_i}{\dot{R}_i}\dot{D}_i \tag{5}$$

Alternatively, as the objects are static, the velocity of the camera can be expressed by (6) using the temporal change of D_i.

$$v = -\dot{D}_1 = -\dot{D}_2 = \cdots = -\dot{D}_i = \cdots = -\dot{D}_n \tag{6}$$

From (5) and (6), we obtain (7):

$$D_i = \frac{R_i}{\dot{R}_i}v \tag{7}$$

From (7), we can obtain the sum of the distances from the visible characteristic points (p_{k+1} to p_m), and thus, we obtain (8):

$$\sum_{i=k+1}^{k+m} D_i = v\sum_{i=k+1}^{k+m}\left(\frac{R_i}{\dot{R}_i}\right) \tag{8}$$

From (1) and (8), we obtain (9):

$$v = \frac{\bar{D}}{\frac{1}{m}\sum_{i=k+1}^{k+m}\frac{R_i}{\dot{R}_i}} \tag{9}$$

In this paper, we utilize a predefined estimated value $\hat{D}$ instead of the actual average $\bar{D}$. Therefore, the estimated velocity of the camera $\hat{v}$ is given by (10):

$$\hat{v} = \frac{\hat{D}}{\frac{1}{m}\sum_{i=k+1}^{k+m}\frac{R_i}{\dot{R}_i}} \tag{10}$$

We conduct simulations using this method in the next section.

V. SIMULATION

We conduct simulations to discuss the accuracy of the proposed estimation method. In the simulations, we set all the characteristic points on horizontal planes (z = 0) to simplify the environments. Figures. 6–9 show the simulation environments. The small circles denote the characteristic points and the triangle denotes the visible area.

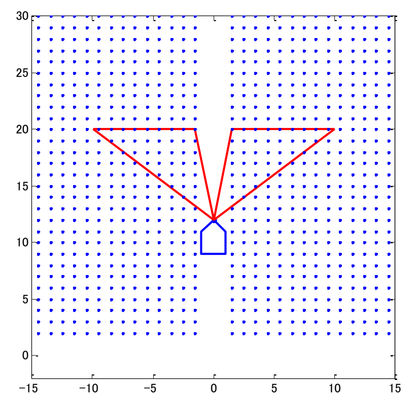


Figure 6. Environment 1 (regular pattern, 900 points).

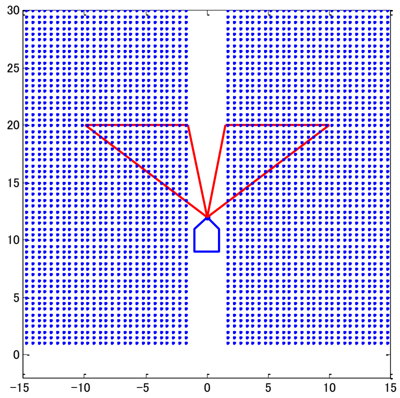


Figure 7. Environment 2 (regular pattern, 3600 points).

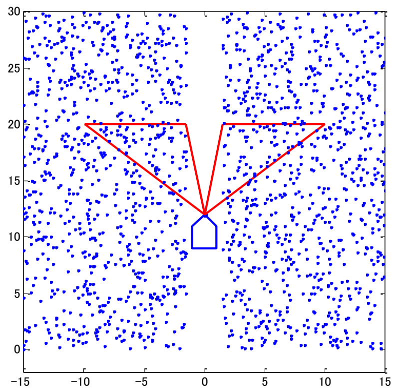


Figure 8. Environment 3 (random pattern, 900 points).

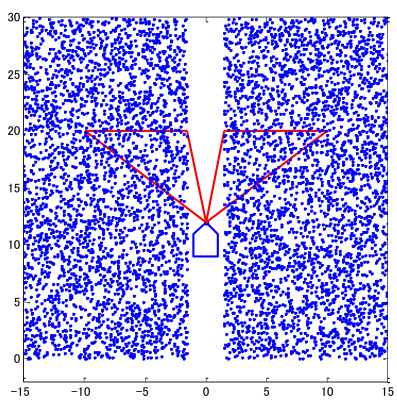


Figure 9. Environment 4 (random pattern, 3600 points).

The characteristic points are set at a regular interval in Environment 1 (Figure. 6) and 2 (Figure. 7), and are set randomly in Environment 3 (Figure. 8) and 4 (Figure. 9). The density of the characteristic points is the same in Environment 1 and 3, and in Environment 2 and 4.

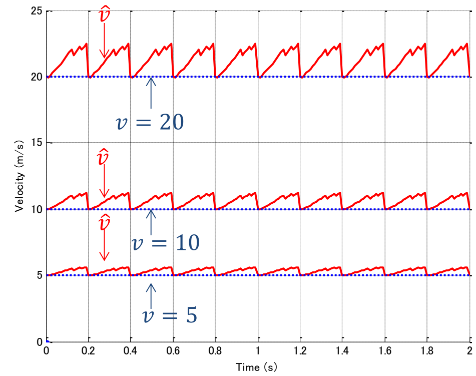


Figure 10. Results of Environment 1

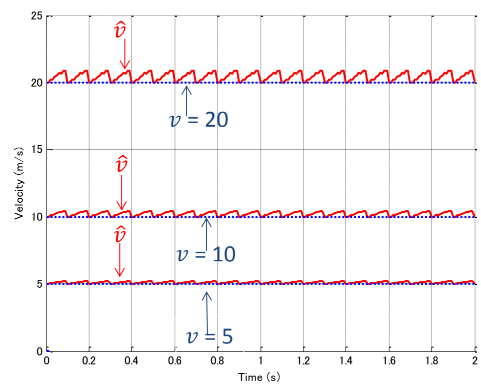


Figure 11. Results of Environment 2

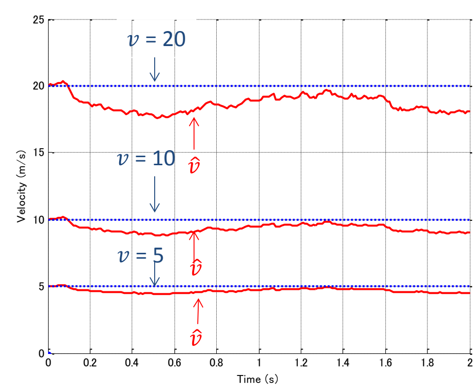


Figure 12. Results of Environment 3

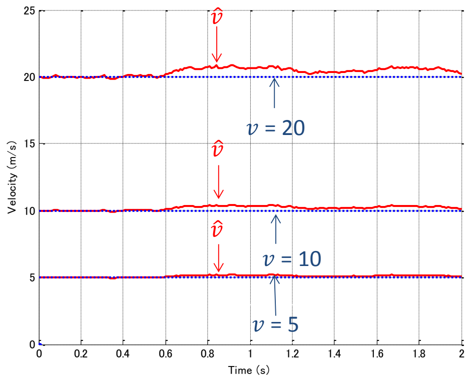


Figure 13. .Results of Environment 4

We conducted simulations at different velocity (v = 5, 10, 20) in every environment. Figure. 10–13 show the results, and based on them, we can infer that the velocity can be estimated. However, the estimated velocity has a small oscillating error. In a precise sense, the average distance $\overline{D}$ is not constant and it oscillates because, while the camera is moving, some characteristic points go out of view and others come into view. At the same time, the sum of $R_i/\dot{R}_i$ also oscillates, and it counterbalances the oscillation of $\overline{D}$. Thus, the actual v is determined as a constant value, as shown in (9).However, as shown in (10), we employ a constant $\widehat{D}$ instead of $\overline{D}$ in the simulations because the actual $\overline{D}$ is unknown. As $\widehat{D}$ is constant and the sum of $R_i/\dot{R}_i$ oscillates, the estimated velocity $\hat{v}$ also oscillates, which causes the error. Therefore, the amount of error depends on the variation of $\overline{D}$. And the error rate of $\hat{v}$ is the same as the error rate of $\widehat{D}$. In the environments, there are a large number of characteristic points and they are uniformly distributed. Hence, the error becomes small, as shown in Figure. 12 and 13.

In summary, we can conclude that the velocity can be estimated using only one camera, and the error rate of the estimated velocity is dependent on the error rate of $\widehat{D}$ and is independent of the actual velocity. As the variance of the error of distance is dependent on the density of characteristic points, the variance of the error of velocity is also dependent on the density of characteristic points.

VI. CONCLUSION

In this paper, we addressed a method for the velocity estimation of a robot from visual information. We focused on the property of the environment and assumed that the average distance from the visible characteristic points was constant and predefined. Under this assumption, we proposed a method for velocity estimation. Simulations were conducted and the results showed that the velocity could be estimated with a small error, which depended on the environment.

Our future work is to develop a prototype system and conduct experiments in real environments, and discuss effectiveness of the proposed framework in practical use.

Acknowledgment

This research was partially supported by the Japan Society for the Promotion of Science through the Grant-in-Aid for Scientific Research (C) 24500181.

REFERENCES

[1] "How Google's Self-Driving Car Works – IEEE Spectrum".Spectrum.ieeee.org, February 26, 2013.

[2] M. Montemerlo, S. Thrun, H. Dahlkamp, D. Stavens, and S. Strohband, "Winning the DARPA grand challenge with an ai robot," in AAAI Conference on Artificial Intelligence, vol. 1, pp. 982-987, 2006.

[3] U. Ozguner ,S. Christoph and R. Keith, "Systems for safety and autonomous behavior in cars, " The DARPA Grand Challenge experience," Proceedings of the IEEE 95.2: pp.397-412, 2007.

[4] S. Thrun, "Robotic Mapping: A Survey", CMU Technical Report, CMU-CS-02-111, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 2002.

[5] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM, "A Factored Solution to the Simultaneous Localization and Mapping Problem, " Proc. AAAI National Conference on Artificial Intelligence (Edmonton 2002), pp. 593-598, 2002.

[6] A. Diosi, and L. Kleeman, "Laser Scan Matching in Polar Coordinates with Application to SLAM," IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS 2005), pp.3317-3322, 2005.

[7] M. Bosse, P. Newman, J. Leonard, and S. Teller, "Simultaneous Localization and Map Building in Large-Scale Cyclic Environments Using the Atlas Framework," The International Journal of Robotics Research, Dec. 2004, vol. 23, 12, pp. 1113–1139.

[8] A. Nguyen, N. Martinelli, Tomatis, and R. Siegwart, "A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics," Int. Conf. Intelligent Robots and Systems, pp. 1929–1934, 2005.

[9] J. Gibson, "The ecological approach to visual perception," Houghton Mifflin, 1979.

[10] J. Gibson, "The perception of the visual world," Houghton Mifflin, 1950.

[11] W. Epstein, and R. Sheena, eds. "Perception of space and motion." Academic press.pp.263-325, 1995.