14TH MIDDLE EASTERN SIMULATION & MODELLING MULTICONFERENCE

MESM'2014

4TH GAMEON'ARABIA CONFERENCE

EDITED BY

Marwan Al-Akaidi

FEBRUARY 3-5, 2014

Muscat, Oman

A Publication of

EUROSIS-ETI

Printed in Ghent, Belgium

14TH MIDDLE EASTERN SIMULATION & MODELLING MULTICONFERENCE

4TH GAMEON-ARABIA CONFERENCE

FEBRUARY 3-5, 2014

Organised by

ETI

Sponsored by

EUROSIS

AOU

Ghent University

IEEE UKRI - SPC

University of Skovde

INCONTROL
Simulation Solutions

Hosted by

AOU – Arab Open University

Muscat, Oman

INTERNATIONAL PROGRAMME COMMITTEE

INTERNATIONAL PROGRAMME COMMITTEE

Simulation in Logistics, Traffic, Transport and Harbour Simulation
Isabel Garcia Gutierrez, Universidad Carlos III de Madrid, Leganes, Madrid, Spain
Daniel Krajzewicz, GermanAerospace Centre, Berlin, Germany
Herve Manier, Universite de Technologie de Belfort-Montbeliard (UTBM), Belfort Cedex, France

Web based Simulation
Jorge Marx Gomez, Otto-von-Guericke-Universitaet Magdeburg, Magdeburg, Germany
Alfonso Urquia, UNED, Madrid, Spain

Simulation in Information Processing
Khalil Drira, LAAS-CNRS, Toulouse Cedex, France
Jean-Claude Hochon, GFI Consulting, Toulouse, France
Peter Lawrence, Deakin University, Bentleigh, Australia
Jim Torresen, University of Oslo, Oslo, Norway

Computational Cognition and Learning
Oussama Hamid, Arab Open University (AOU) - Kuwait Branch, Al Ardia, Kuwait

Simulation in Fuzzy Systems, Neural Networks and Genetic Algorithms
Track Chair: Kifaya Qaddoum, Warwick University, Coventry, United Kingdom
Chrissanthi Angeli, Technological Education Institute of Piraeus, Athens, Greece
Viorel Nicolau, University of Galati, Galati, Romania

Simulation in the Built Environment
Bernadette O'Regan, University of Limerick, Limerick, Ireland
Agustin Yague, Universidad Politicnica de Madrid, Madrid, Spain

Simulation in Archaeology
Jean Yves Blaise, UMR CNRS/MCC 694 MAP, Marseille, France
Anders Hast, University of Gavle, Gavle, Sweden
Wolfgang Kreutzer, University of Canterbury, Christchurch, New Zealand
Marcos A Rodrigues, Sheffield Hallam University, Sheffield , United Kingdom

Simulation Projects in the Arab World
Saleh Al-Jufout, Tafila Technical University, Tafila, Jordan
Mohamed-Aly O. Louly, University of Technology, Troyes, France

GAMEON-ARABIA
Pieter Jorissen, Karel de Grote Hogeschool, Hoboken, Belgium
Ian M Marshall, Coventry University, Coventry, United Kingdom
Joseph M. Saur, GTRI, Joint Systems Integration Center, USA

MESM’2014

GAMEON-ARABIA’2014

All articles published in these Proceedings have been peer reviewed

EUROSIS-ETI Publications are ISI-Thomson and IET referenced

A CIP Catalogue record for this book is available from the Royal Library of Belgium under nr.12620

For permission to publish a complete paper write EUROSIS, c/o Philippe Geril, ETI Bvba, Executive Director, Greenbridge Science Park, Ghent University-Ostend Campus, Wetenschapspark 1, Plassendale 1, B-8400 Ostend Belgium.

EUROSIS is a Division of ETI Bvba, The European Technology Institute, Torhoutsesteenweg 162, Box 4, B-8400 Ostend, Belgium

Printed in Belgium by Reproduct NV, Ghent, Belgium
Cover Design by Grafisch Bedrijf Lammaing, Ostend, Belgium

EUROSIS-ETI Publication
ISBN: 978-90-77381-81-6
ISN: 978-90-77381-81-6

Preface

Dear conference delegate,

I have the pleasure to welcome you to the combined 14th Middle Eastern Modeling & Simulation Multiconference (MESM2014) and 4th annual Pan-Arabic - GAMEON-ARABIA'2014 organized by EUROSIS and hosted and sponsored by the Arab Open University (Oman Branch) in Muscat, Oman. The MESM'2014-GAMEON-ARABIA'2014 is co-sponsored by the IEEE – UKRI SPC, Incontrol Simulation Solutions, Ghent University, and The University of Skovde.

While the MESM'2014 Conference highlights recent and significant advances in many research areas of modeling and simulation related to decision support systems, electronics simulation, network simulation, engineering simulation and video technology, GAMEON-ARABIA'2014 looks at the advances in gaming research in the Middle East.

Next to the programme, featuring the refereed and selected papers, the joint event also features an excellent keynote by Assoc. Prof. Björn Johansson of Chalmers University of Technology, entitled "Utilizing Simulation for a More Sustainable Production".

As General Conference Chair of both events, I would like to express my thanks to Professor Moudi Al Hmoud, the Arab Open University Rector, for giving me the time to chair this conference and thanks also to the committee members for reviewing the papers and to our local chair Dr. Moosa Al-Kindi, Oman Branch Director of Arab Open University in organizing this event at our Oman Branch.

Thanks to my colleague Philippe Geril, executive director of EUROSIS office for supporting the event and for his time. Last but not least thanks to all authors without whom the conference would not be a successful conference.

Professor Dr Marwan Al-Akaidi
General Conference Chair
EUROSIS – M. East Chair
Vice-Rector Academic Affairs
Arab Open University
HQ- Kuwait

CONTENTS

CONTENTS

SCIENTIFIC PROGRAMME

DECISION SUPPORT SYSTEMS

A TYPOLOGY OF STAKEHOLDER IDENTIFICATION METHODS FOR PROJECTS IN SOFTWARE INDUSTRY

Smiju Sudevan
Department of Computer Science
Modern College of Business and Science
Muscat, Oman
smiju@mcbs.edu.om

M.Bhasi
School of Management Studies
Cochin University of Science and Technology
Cochin, India
mbhasi@cusat.ac.in

K.V.Pramod
Department of Computer Applications
Cochin University of Science and Technology
Cochin, India
pramodkv4@gmail.com

KEYWORDS
Stakeholder, Software management, Stakeholder management, Stakeholder identification, Stakeholder analysis

ABSTRACT
There are many reasons behind the failures of a software project. Failure is strongly related to stakeholder's perceptions of project value and their relationship with the project team. A project is more likely to be successful if it begins well. A good beginning includes time at the outset to discuss relevant software project stakeholder's key needs and expectations. Despite the enormous growth of participatory software practice and theory, there is still little shared understanding amongst those involved. The stakeholder structure in software is complex and dynamic and it is imperative to identify all the relevant stakeholders involved. In this respect, Stakeholder Identification is a crucial step through which the different expectations and influences can be assessed. This paper intends to dwell upon the significance of stakeholder identification in software industry by doing a literature study.

I. INTRODUCTION

Software project management has proved to be one of the most difficult tasks in software development business. As software development projects adapt to changing circumstances, management of those projects must also adapt. Such changes could involve adopting different development lifecycles, moving to component based software development or distributing the development. The knowledge of the mechanisms could guide the ways in which project managers adapt to different project environments as well as guide efforts to develop project work flow and project situation. The software literature provides many published papers and surveys[1][2] that report alarming figures for software project failures at various levels or discuss inconvenient project situation[3]. Project failure is at times associated with poor decision making and lack of stakeholder involvement in searching for solutions to project management problems[8].Stakeholders are the first emerging challenge in any software project. The importance of stakeholders during the software lifecycle process is based on the premise that their activities in the initial stages largely determine the quality of the finished software product and as such stakeholders yield power and influence over the project manager. It is seen that when a project is first communicated and addressed to its stakeholders, this will greatly influence the likelihood of its success [4]. That is, adequate stakeholder participation is essential in software just like any other industry. The stakeholder structure in the software industry is complex and dynamic, while the product can have pervasive effects on many industrial sectors and wider segments of society. In this respect, it is critical to identify all the people and organizations who may have an impact on the project, and all those who are impacted by the same very early in the life of the project. However, the identification of stakeholders as well as their needs and expectations is poorly done in software projects, probably because this process is mistakenly viewed as a self evident task in which the direct users, clients and the development team are the only stakeholders [12]. This paper intends to underline the significance of stakeholder identification in the management of software projects. For this, no special purpose method is necessary to achieve the objective of this research. Various findings pertaining to the relevant literature has been reviewed to help determine the effectiveness of stakeholder identification methods or techniques.

II. STAKEHOLDER IN LITERATURE

Stakeholders are individuals and organizations "who are actively involved in the project, or whose interests may be positively or negatively affected as a result of project execution or successful project completion" (Project Management Institute, 2011).According to Hitt, Freeman and Harrison (2001) the use of the term stakeholder emerged in the 1960s from pioneering work at Stanford Research Institute, which argued that managers "needed to understand the concerns of shareholders, employees, lenders and suppliers, in order to develop objectives that stakeholders could support". The term has become increasingly prevalent since Freeman's (1984) seminal text "Strategic Management: A Stakeholder Approach". The global meltdown of financial markets and widespread corporate collapses of 2008 re-focused public debate sharply on questions of the relationship between business and society and the design of the corporation of the future –

"shifting the purpose of the firm to encompass not just shareholder needs but also societal, stakeholder and ecological needs and interests" [6]. The business benefits of effective stakeholder engagement are now well-known and well- documented. A number of studies have found a clear correlation between stakeholder relationship quality and financial performance [4]; sustainable wealth/long-term value [7] and corporate reputation [6]. The stakeholder approach was found to be a strategic management tool[5] - instrumental as opposed to normative .The emergence and establishment of a social performance agenda for business has highlighted the value of stakeholder theory as a "normative approach that some argue is more ethically and morally acceptable than a shareholder value approach" [5].Since the mid 1990's,this question of the legitimacy of stakeholder claims on organizations has emerged as central to the debate relating to corporate social responsiveness and corporate responsibility.Svendsen (1998) argues the case for competitive edge as an outcome of effective stakeholder engagement: "as paradoxical as it sounds, one way to succeed in a highly competitive globalised economy is to co-operate". The central claims for an integrated approach to stakeholder engagement arguably centre primarily on benefits to the organization – essentially on the view that "incorporating stakeholder views in decision-making processes enhances organizational performance and commitment" [8].

III.STAKEHOLDER MANAGEMENT

Stakeholder Management is one of the emerging challenges of software industry. Project management is an important part of software development, both for organizations that rely on third-party software development and for those whose software is developed primarily in-house. . One of the key skills in project management, therefore, is to be flexible and to adapt to any management tools. Software and web services companies have faced new challenges. The adoption of project management is seen as a method for solving such organizational problems [9]. Software Project management is now well developed and well accepted as a domain for the exercise of professional expertise and as an area of academic research and discourse. Some studies had focused on problems in software development projects. Each of these stakeholders has a different reason for having an interest in the software system, which influences their behavior. Management of these "stakeholder interests" is referred to as Stakeholder Management. A clear and explicit representation of the stakeholders and their attributes is required in order to achieve their effective management. While this is important in every project, it is especially important in software development where deliverables are not tangible [9].

The Standish survey shows that less than 50% of software projects can be classified as successful. That is only less than 50% of the projects are completed on time with in the budget and covering the scope. This knowledge was found to be very alarming [1]. It is possible that poor project management knowledge and skills in software companies lead to the project failures [2]. Stakeholders were found to be one of the main determining factors of the above.

IV.RELEVANCE OF STAKEHOLDERS IN SOFTWARE

The success of a software project depends on how stakeholders are handled. The successful management of stakeholders can have a substantial and immediate impact. Satisfied stakeholders can greatly improve the progress and relevance of a project and ultimately contribute significantly to its success[17]. A typical software quality and expectation is the lack of defects. A software product that was thoroughly tested and bug free may not meet current or future stakeholder expectation. Looking at the broader definition of software quality, the project manager in the project described can identify all of the stakeholders- project team, software supplier, user department and the IS department manager[14].Although the term stakeholder is often used in software development, it is predominantly limited to customers, end users or project sponsors.

The role of stakeholders in the software firm's strategy and operation has received considerable attention in management research over the past two decades. The reasons for advocating the stakeholder approach fall into two broad categories. First, there is the normative argument that the firm has the moral obligation to account for its activities towards those affected by them. The second argument is more pragmatic: firms that take into account the needs and interests of their various stakeholders are financially more secure and successful, according to stakeholder theorists (Beaver, 1999). The entire process of software project management is strongly stakeholder-driven. It is their stakes, that determine the course of the project. A team will have to deliver a project under time pressures to appreciate the constructive power of motivated people or the destructive power of demotivated team members. In a project, it is the people that are the main cause of problems. Time schedules, financial projections, and software goals may be abstractions, but it is the flesh-and-blood people whose work determines the project's status.

V. STAKEHOLDER IDENTIFICATION

Stakeholder identification plays a key role in the stakeholder management effort. The stakeholder identification process is the very first step taken in initiating a new project [10]. It is a precursor to stakeholder analysis report. Only if stakeholder identification is done initially, can the requirements of the project can be elicited [9]. Thorough stakeholder analysis firstly requires the identification of the relevant parties,that is, primary stakeholders have to be distinguished from less influential participants .A typical software engineering project team consists of a project manager, business analysts, developers, testers, and quality assurance personnel. The team will generally include users or their representatives. Unfortunately, the literature on software engineering and project management often fails to clearly classify types of stakeholders and to describe strategies for their management. For example, Mitchell gave a list of 27 definitions of the term stakeholder used from 1993 to 1995, showing the intention of researchers to answer the fundamental question of which individuals can be defined as stakeholders that deserve the manager's attention [9]. A stakeholders' significance will depend upon the situation and the issues encountered during the project lifecycle. Of all the possible stakeholders the ones who will be relevant to the organization's management team will depend on the particular project. The project size and the project complexity, directly affect problem levels and, as such, the breadth and depth of stakeholder involvement.

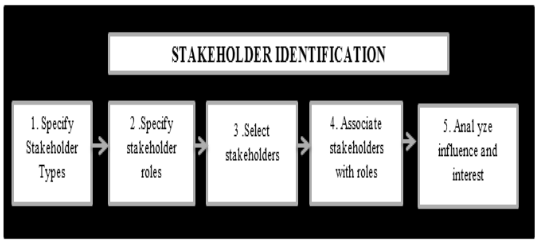


Fig 1: Levels of Stakeholder identification

Each software project includes different types of stakeholders, so it is necessary to identify, characterize and handle all the viewpoints of the different types of stakeholders [18]. This may vary from project to project. There are different types of stakeholders who need to be identified in each project.

1. In an SEI report[10], stakeholders identified come from at least five communities involved in software development; clients/ sponsors, users, developers, quality personnel, security personnel and requirements analyst. In each of the above said areas, the different stakeholders involved are classified and their different roles are defined on the basis of activities performed.

2. Ian Somerville [10] places the identification of stakeholders within the stages of obtaining and analyzing the software requirements. Among the identified stakeholders are the final users who will interact with the system. Stakeholders also include engineers who develop or support other related systems, for example, business managers, IT specialists, workers representatives etc.

3. Roger Pressman[15] argues that stakeholders must be identified in the beginning of the requirement engineering process because many different participants are involved at the state.Pressmann identifies the following stakeholders as the most common business managers, brand managers, marketing staff, external and internal customers, consultants, product engineers, support and maintenance engineers etc.

4. In the Rational Unified Process [16], within the software engineering process stakeholder identification is carried out at the management, requirement processing stage. The most obvious stakeholders in a software project are the final user, software developer, purchaser, project director, anyone strongly interested in the project or those who need the project to solve their needs.

5. The Organization patterns identified by Coplien and Harrison directly addresses specific stakeholders, and the relationship between stakeholders in agile methodology [19].

Identifying stakeholders is necessary to ensure complete, correct requirements specification. In addition, we must identify and mitigate the risks they introduce to the project. Stakeholders can introduce risk in all software project phases. The requirements phase is particularly vulnerable because the project team will interact directly with many stakeholders to define requirements. When stakeholders have a negative perception of the project, they will likely be less open, honest, thorough, and helpful, resulting in incomplete, incorrect requirements. This in turn will have a big impact on final software product quality [2].

VI. TOOLS AND TECHNIQUES OF STAKEHOLDER IDENTIFICATION

The first step concerned is with the phase "Who are the stakeholders", Stakeholders are all those who need to be considered in achieving project goals and whose participation and support are crucial to its success. Stakeholders can be 1) individuals within the project 2)individuals and groups within the organization and 3)individuals or groups outside the organization. Thus, there are many software stakeholders to be identified for a project.

Stakeholder identification starts usually with identifying more relevant stakeholders and move towards those who interact with them. Freeman[4,5] identifies 7 techniques for creating value for stakeholders, including stakeholder assessment, stakeholder behavior analysis, understanding stakeholder in more depth, assessing stakeholders strategies ,developing specific strategies for stakeholders, creating new models of interaction for stakeholders, developing integrative value creation strategies.

Alexander and Stevens give a practical guide for identifying stakeholders [10], which starts with the identification of leaders. Here, managers assist the project team in listing the people who should be involved, according to decision levels and hierarchies existing in the organizational structure. Sharp etal [14] proposed a methodology where the first step is to define a stakeholder's baseline formed by stakeholders groups (such as users, developers, decision makers etc). In further steps, they evaluate who the suppliers, clients etc are and also who all interacts with each group given in the baseline. In this view, identification is concluded when all the groups in the baseline are analyzed. Similarly, Robertson [13] and Robertson and Alexander [17] presents a well explained model describing diverse stakeholder types using the onion model and locating each type of one of the onion levels (rings).They work with producers, consumers, sponsor's, influencers and consultants and others as stakeholder types .They explain how each type must be identified in the model, and the people to be included in each concentric circle.

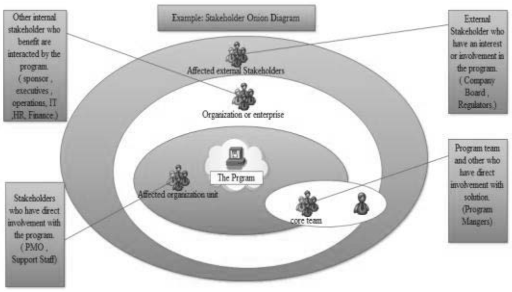


Fig 2.Onion diagram representing stakeholder roles

The following are the different ways to identify stakeholders who play a role in the software project. In doing this, the main objective is to try to break the large group of stakeholders into smaller groups since large groupings can impact the value of information gained from the process.

1) Category Approach: It is the most commonly used method where categories of stakeholders are created by the project team based on past experience and these are then used to identify stakeholders. The risk of this approach is that it may be too broad resulting in overlooking of some stakeholders.

2) Role Approach; The project team works from a generic list of stakeholder roles. Because the roles are very generic, this approach makes it easy to overlook stakeholders who don't have a direct interactive role in the system or project.

3) Interview Approach: The interview method is most useful for identifying new stakeholder roles and new individuals to potentially fill those roles. Unfortunately it is time consuming and is found to be unfeasible in majority of software projects.

It can be seen in some cases that the above methods may produce a really long list of stakeholders which may be very extensive. It's not necessary that all the stakeholders need to put time and effort in the software project activities. Full stakeholder participation will also cause the process to be complex and time consuming, which should be avoided if possible. The level and participation and the effort put into stakeholder involvement can be adapted according to their positions [13].

Stakeholders can be categorized by the amount of influence they can have over the project success. This means not only identifying the level of influence the stakeholder has within the business overall, but also the level of influence they may have on the projects budget, access to resources, implementation, persuasive influence over key decision-makers, or control of critical knowledge[14]. Indeed, the project team may want to map each stakeholders influence level across multiple aspects of the project space (such as budget, resources, knowledge) in order to more fully understand their potential impact as well as when to best engage with each stakeholder. This assessment leads to strategies for communication and strategies for assessing stakeholder satisfaction, it culminates in the development of a stakeholder knowledgebase that provides knowledge of who is aware or ignorant and whether their attitude is supportive or opposing (Turner 2002).

The broad view of stakeholder identification focuses on stakeholder ability to influence and organizations behavior, direction, process or outcome, and focuses on the urgency, power, and legitimacy of the stakeholder in question. For this stakeholder positions we use the stakeholder typology developed by Mitchell et al (1997) [9]. What sets this typology apart is that it is not based on characteristics of the stakeholders themselves, but on determinants of stakeholder salience (Mitchell et al 1997).Some stakeholders will naturally have a higher degree of relative importance or urgency.Saleince, the principle of defining who and what really counts to an organization, is the key to understanding the stakeholders that project manager should pay most attention to. Here, three variables in the stakeholder-firm relationship that determine stakeholder salience, ie, power, legimitimacy and urgency are taken into account. According to above, eight types of stakeholders are identified. Power describes the ability of an individual to make another individual to do something which he otherwise would not have done. Legitimacy is the degree to which the firm and the stakeholders find each other's actions desirable, proper, appropriate etc. Urgency is the degree to which the stakeholder claims immediate attention.
(Mitchell et al 1997)
The eight types of stakeholders are

1. Dormant stakeholder
2. Discretionary stakeholder
3. Demanding stakeholder
4. Dominant Stakeholder
5. Dangerous Stakeholder
6. Dependent Stakeholder
7. Definitive Stakeholder
8. Non- stakeholder

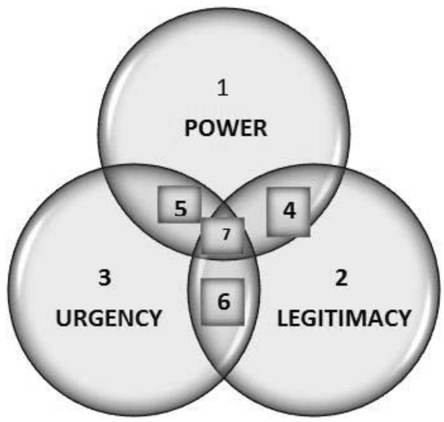


Fig 3: Qualitative classes of stakeholders

The results of the Stakeholder Identification effort should include a complete list of stakeholder roles as possible given the knowledge at that time, the above information is classified and put in the stakeholder registry. Though initially most of the software companies did not deem it necessary to have a registry, changing times have prompted most of them to give due importance for stakeholder identification and putting up a registry in the initial phases of the project itself.

VII.CONCLUSION

Stakeholder management thus enables in managing stakeholder's expectations and ensuring their active involvement in the project from the initiation stage itself. There seems to be an assumption that Stakeholder Identification is an easy and relatively straightforward process. In reality, achieving a complete stakeholder list seems to be anything but simple. It must be remembered that the purpose of the Stakeholder Identification process is to identify all stakeholders, not which specific stakeholder will best fit a specific stakeholder role. It is therefore of great interest to understand and predict likely problems with project social networks and corresponding stakeholder involvement at the start of the project, so that potentially disastrous problems can be mitigated. This paper thus wants to review that stakeholder identification is crucial for the success of software projects and it is dynamic. Also, at this stage, the present paper does not represent a uniform technical framework. Instead, it is seen that the different processes should be tailored for stakeholder identification based on the requirements for the software project. And it is to be said that the approaches described in this paper are based on qualitative proposals where Stakeholder's attributes were evaluated through subjective recommendations.

REFERENCES

[1]Standish 1994, Standish group International. The Chaos report.

[2]El Emam , K.Koru A G(2008). A replicated survey of IT software project failures. IEEE software 25(5)84-90.

[3]Boehm B, 2000.Project termination doesn't equal project failure. Comp 3(9) 94-96.

[4] Hitt, M., Freeman, R. & Harrison, J. (2001). The Blackwell Handbook of Strategic Management. Hoboken, NJ: Wiley- Blackwell Publishers Ltd.

[5]Freeman, R.E. (1984). Strategic Management: A Stakeholder Approach. Boston: Pitman

[6]Cooper, SC (2004). Corporate Social Performance: A Stakeholder Approach. England: Ash gate Publishing Limited.

[7]King, A. (1998). The social performance uncertainty principle. Corporate Reputation Review, v 1, no 3, pp 43-46. Henry Stewart Publications.

[8]Post, J., Preston, L., & Saschs, S. (2002). Redefining the Corporation. Stakeholder Management and Organizational Wealth. Stanford Business Books. California: Stanford University Press.

[9] Ronald K. Mitchell, Bradley R. Agile, Donna J. Wood, (1997).Toward a Theory of Stakeholder Identification and Salience: Defining the principle of Who and What Really Counts, The Academy of Management Review,pp. 853-886

[10]Ken Power,(2010). Stakeholder Identification in Agile Software Product Development Organizations - A model for understanding who and what really counts, IEEE Publication, pp 87-94, AGILE Conference

[11]Svendsen A. (1998) The Stakeholder Strategy. Profiting from Collaborative Business Relationships. San Francisco: Berrett-Koehlert Publishers Inc

[12]Lim, S.LBentley, P.J., 2011.Evolving relationships between social networks and stakeholder involvement in software projects.In: Proceedings of the 13th Annual conference on Genetic and Evolutionary computation, Doublin, Ireland, pp.1899-1906.

[13]Robertson S (2000) project, sociology: Identifying and involving stakeholders The Atlantic system guild .www.system guild.com

[14]K.Power," Software Development: A Stakeholder Approach (PhD Thesis),"J.E.Cairnes School of Business & Economics, National University of Ireland, Galway, In Progress.

[15]Pressman Roger S.Software Engineering. A Practitionre's approach 6th edition McGraw Hill 2005

[16]Krutchen, Phillepe, The Rup an introduction 3rd edition Addison Wesley, 2003

[17]Alexander I, Robertson S (2004) Understanding project sociology by modeling Stakeholders IEEE Software, IEEE computer Society 21(1):23-27

[18]Pacheco C,Tovar E(2007) Stakeholder Identification as an issue in the improvement of Software Requirements Quality.Krogstie J,Opdahl AL,Sindre G(eds) CAiSE 2007,LNCS 4495,pp 370-380

[19]J.O.Coplien and N.Harison.Organizational pattern of agile software development, upper saddle river, NJ, Parson 2004

AQUASCOPE: AN ONLINE-BASED DISTRIBUTED WATER QUALITY ANALYSIS METHODOLOGY USING AGENTS

N.R Wilfred Blessing
Shajulin Benedict
HPCCLoud Research Laboratory,
St. Xavier's Catholic College of Engineering,
Affiliated to Anna University, Nagercoil 629003, India.
E-mail: wilfred_blessing@yahoo.co.in
shajulin@sxcce.edu.in

KEYWORDS

Water Quality Analysis, Distributed Artificial Intelligence, Intelligent Environmental Monitoring, Agents, Water Quality Parameters.

ABSTRACT

Water Quality Analysis is one of the essential steps to form a healthy environment. The analysis when combined with the intelligent computing technologies could support the environmental analysts, agricultural planners, market trend predictors, or the water quality control officials to make a prompt decision about using the water or the representational land within a reasonable time. In order to do the analysis, the water quality should be sensed at various sites and computed periodically for better results - the analysis is data-intensive and compute-intensive. The conventional water quality analysis procedures are either offline, manual, or analyzed centrally. In this paper, we have proposed an online-based distributed water-quality analysis tool named AquaScope. The proposed toolkit collects the water quality sensor data from user-specified geographical locations, analyses the quality of water, and reports on the severity of water quality automatically.

INTRODUCTION

Quality water is one of the solutions for framing an economically healthy world. Waters are gifted in various forms such as, rivers, lakes, canals, drains, sea and ground waters. Urbanization, industrialization, and improper waste management, however, have challenged living beings due to the water prone diseases. Despite, Water quality Analysis (WQA) is an important aspect to framing an economically healthy world.

Water is used for drinking, irrigation, industries, aquatic life, fisheries, hygiene purposes (washing and flushing toilets), recreation (swimming and boating), gardening and for other domestic purposes. Quality is differed based on its usability and utility. Drinking water should be pure when for the other use it may be impure in few aspects. Water Environment Management or the Water Resources Protection System provides an effective platform to improve the customary standard which satisfies the water quality needs. Environmental experts and AI scientists are responsible to present consistent models for water quality (WQ) management techniques, continuous monitoring plans and competent treatment schemes.

WATER QUALITY ANALYSIS

Informational technologies have played a key role in planning, prediction, supervision and control of environment processes at many different scales and in various time periods (Cortes et al. 2000). There are numerous WQ factors determined for finding the quality of the water. Commercial sensors are available to provide data simultaneously for the list of parameters given in Table 1. Water with higher concentrations of certain parameters can represent a significant risk on human health and aquatic organisms. Diarrheal diseases are caused due to unsafe drinking water. Children are affected in huge percentage due to poor water quality. It is necessary to investigate appropriately and create proper awareness for the public before the water is used from different water catchments or water sources. In this paper, the quality of different water sources are analyzed at various spots. The proposed system structure is named, locally as "The Limited site" which compute WQA results using sensor data and globally as "The Distributed site" that generates WQA outputs using dissimilar water quality input formats.

WATER QUALITY VARIABLES

Water Quality parameters are determined based on the physical, chemical and biological characteristics of water. A variety of water quality parameter variables are available. The selection of parameters depends upon the intentions. Appropriate selection of parameters will help the intentions to achieve, powerfully and in most cost effective way.

Table 1: Selected water quality variables

Water Quality Parameters	
Temperature	Turbidity
Dissolved Oxygen	pH
Conductivity	Salinity
Depth	TDS
Total Dissolved Gas	Nitrate
Chlorine	Chlorophyll
Rhodamine WT	ORP
Specific Conductance	Ammonium
Blue Green Algae	Ambient Light (PAR)

BACKGROUND OF DISTRIBUTED ARTIFICIAL INTELLIGENCE

Water sources are spread throughout the world. In our proposed scheme, the distributed approach is used. A middleware tool is considered as primary-agent and the data service providers (DSP's) as sub-agents. Distributed artificial intelligence (DAI) models resolve problems using agents. Here the control and information are frequently distributed among the agents. These agents operate in parallel and increases problem solving speed and decrease the complexities of each agent. Multiple agents learning resolve large complex problems, increases domain knowledge, communication knowledge, and improve performance of group of agents. Therefore, agents perform communication in most efficient manner. Each agent possesses learning and modeling the knowledge (Zhongbo et al. 2010). Agents are capable to solve many type of jobs or services. Service allocation is simple in distributed AI systems. Primary-agent should communicate with sub-agents for reliable results.

NEED FOR ONLINE ANALYSIS

Commonly, the data are represented by means of online, offline or manual. People around the earth are connected worldwide with internet technologies. Continuous monitoring and intelligent decision making support using cloud systems (Benedict 2013) are considered necessary for environmental scientists. Improved system performances with fast, reliable results are desired by the users. Monitoring anytime in the world is essentially required for efficient transformation and examination of an outcome. A well suited and equipped multi-parameter optical sensors are capable of investigating and producing online analysis of WQ.

EXISTING ANALYSIS METHODS

Existing Water Quality software models such as SWQAT (Sharma et al. 2013), CALHIDRA (Cardona et al. 2011), WaterRAT (McIntyre and Wheater 2004), SWAT (Coffey et al. 2010), MCAT (Marinoni et al. 2009), DUET-H/WQ (Harmel et al. 2009), SQA (Khosrovyan et al. 2010), E2 (Argent et al. 2009), EVS (Brown et al. 2010), mDSS (Giupponi 2007), IRA-WDS (Vairavamoorthy et al. 2007) etc., are presented by various environmental computer scientists in last 10 years.

Tools like AQUATOOL, MODSIM, RIBASIM, WARGI-SIM and WEAP models are considered in a preliminary analysis that considers alternative plans and policies in water system (Sulis and Sechi 2013). Embedded web technology and CAN bus technology are used in the online water quality monitoring platform. This system collect WQ parameters in many sites, displays in web browsers and provides early warning. It replaces the traditional proxy project based on PC (Zhongbo et al. 2010). OIP (Overall Index of Pollution) technique is used for water quality assessment using 14 WQ parameters for a tool named SWQAT - Surface Water Quality Assessment Tool (Sharma et al. 2013). Out of 14 specified parameters, only a small number of the parameters are considered mostly to find the quality results. Distributed approach and WQ sensors were not utilized to collect WQ information. In many cases, WQ analysis was done in offline-mode and only for single source of water at various time span. AI (Artificial Intelligence) techniques like fuzzy models, multi-agent system, reinforcement learning techniques, hybrid systems etc., are described for environmental modeling.

SYSTEM STRUCTURE

Overview of AquaScope

AquaScope is an online WQA toolkit, based on a distributed approach and service agents. A number of commercial sensors are potentially suited to the AquaScope project. Multiparameter WQ sensors which can provide results simultaneously in online basis are utilized.

Framework for Limited Site

Based on "intelligent knowledge" in computational models, any resourceful intelligent system includes several processing parts which can be deployed in a distributed information system. AquaScope stands top in performing segment-oriented tasks and agent based performances on services. AquaScope framework for limited site functions are grouped into four segments (figure 1). This Limited site architecture is connected with distributed environment using a middleware tool.

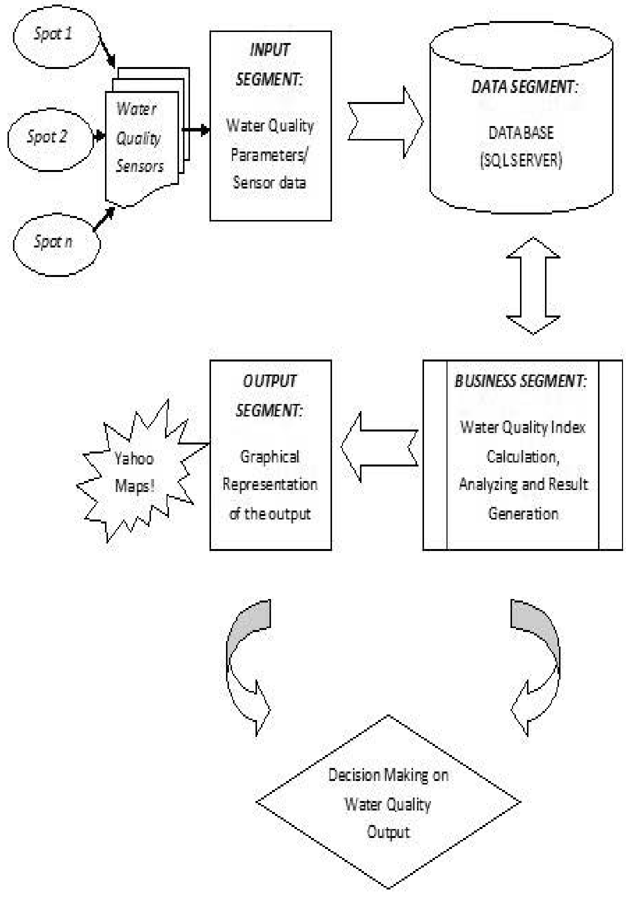


Figure 1: AquaScope framework for Limited Site

Four Segments

Input Segment

WQ parameter values are given as an input to the proposed system towards the input segment. WQ inputs could be entered yearly, seasonally, monthly, or weekly through this segment. Three forms of inputting methods are advocated. They are the sensor input, manual input and input through agents using web services in an distributed environment. However, online based inputs are promoted. Sensor named DS5X multiparameter water quality sonde is used to measure upto 16 parameters simultaneously.

Database Segment

Online inputs of WQ data are stored in SQL database server which is connected using ODBC – Open Database Connectivity to the user-friendly software interface toolkit. Popular water quality results database that are available in various information centers and organizations are utilized for large data analysis. For the limited site, appropriate metadata content principles and data models are identified and the relational database are designed to store all the input parameters needed for the AquaScope project using Microsoft SQL Server 2000.

Business Segment

This is the core system which calculates the water quality indexes, sub-indexes (Sharma et al. 2013) and quality results using an effectual mathematical methods. Based on the selected water quality parameters and specified range of each parameter, water quality indexes for all the entered water quality parameters at a particular spot are calculated in this segment. Various computational models with the business logic are used for proficient solution of expected results. In the distributed site (figure 2), the computations are handled by the primary-agent the middleware.

Output Segment

The results of business segment are visually interpreted graphically and that are observed through user-friendly graphical user interface (GUI). Based on the graphical representation of the results and output, proper decision is taken by the environmental engineers. The quality outputs are viewed in a free online portal Yahoo-maps as hotspots. Through the graphical representation of sensible WQ outputs, environmental engineers and managers are able to make proper decisions for further processing in order to improve the quality of the water at particular source and spot. Also the water treatment plans could be suggested by viewing the generated outputs.

Framework for Distributed Environment

Each Data Service Provider (DSP) acts as an intelligent sub-agent which collects the water parameter information that are distributed worldwide in various forms like web services (eg: input from websites), file formats (eg: XML file), message queue format (eg: messages), data store (eg: DB). Sub-agents call for service requests and wait for responses. Once the distributed data is received by intelligent sub-agents, then the WQ parameter data are integrated and converted using a well organized multi agent system named middleware tool. AquaScope calls middleware and ask for services. Middleware interacts with various sub-agent DSP's for different input formats and provide services to the limited site. A multi agent middleware tool, "IBM Web Sphere Message Broker v8.0 (Achieves ESB and SOA Architecture)" is used for accurate communications. WSDL (Web Service Description Language) the XML based language is used to describe the service businesses and to provide a route to access those services. Distributed site is connected with limited site which generates the WQ output for decisions. The data communication format in distributed environment may be either synchronous or asynchronous type.

In the distributed site, all the water quality computations, transformations and data formatting are calculated by the primary agent, the middleware. Middleware interface code are written in an object oriented programming tool JAVA. ESQL (Extended Structured Query Language) is used to reduce the complexity of all the applications.

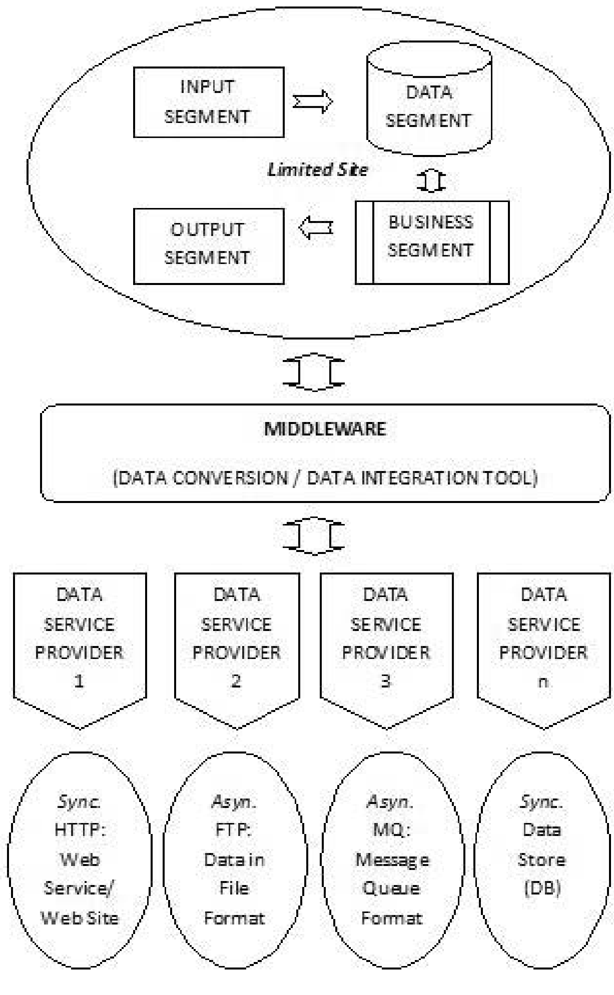


Figure 2: AquaScope framework in Distributed Environment

Water Quality Classification and Results

WQ results are classified as six divisions (Excellent, Acceptable, Slightly Severe, Severe, Very Severe and Extremely Severe) that are grouped into three main sections as given below (figure 3). Based on these classifications, water quality ranges are separated and the severity reports are automatically produced by the AquaScope toolkit. WQ computations for few sampling spots of some water sources are tested on preliminary stage of AquaScope. Water quality of the sampling spots are classified as Average, Polluted and Heavily Polluted. The results are displayed in graphical representations to analyse the feasible reasons of pollution. The resulted values (Spot name, Date of evaluation, Time of analysis, Range of pollution and the WQ result) are displayed in yahoo-maps.

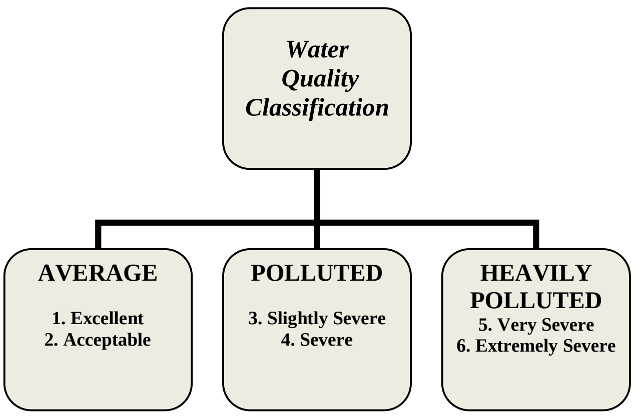


Figure 3: Output Classification of WQ

Advantages and Benefits

AquaScope can handle any kind of data which are distributed around the world. So it is highly compatible and also flexible due to platform independent and technology independent operating facilities. Users can view the water quality information via a user-friendly web browser anytime and anywhere in the world. This saves the cost of data communication and facilitates the sharing of resources which can create the public alertness for the users and environmental experts for further decisions. Primary-agent and sub-agents function in parallel and increases data integration or conversion speed, communication and decreases the complication of each agent.

CONCLUSION

AquaScope is an online-based distributed water quality analysis methodology estimating WQ results for different mode of inputs. The primary focus is to providean user-friendly toolkit capable of providing quick interactions and services through agents for commonly available data globally. Quality of water catchments can also be analyzed in various hydrological conditions in order to avoid seasonal influences, in particular, such as flood and non-flood seasons. WQ in flood season may be better than in non-flood season. Based on the generated results, the system can also suggest some standard water treatment techniques. The design of hardware based web servers implemented with most advanced processors and middleware interfaces may be used to provide reliable communication modes which can be suitable for monitoring quality of water in different environmental situations.

ACKNOWLEDGEMENTS

This work is carried out at the HPCCLoud Research Laboratory (www.sxcce.edu.in/hpccloud) of India. The authors wish to thank the Department of Science and Technology, India and GIZ-CIM, Germany.

REFERENCES

Argent R. M; J.M. Perraud; J. M. Rahman; R. B. Grayson; and G. M. Podger. 2009. “A new approach to water quality modelling and environmental decision support systems”. *Environmental Modelling and Software 24*, 809-818.

Benedict .S. 2013. “Performance Issues and Performance Analysis Tools for HPC Cloud Applications: A Survey”. *Computing*, V-95 No:2, Springer, 89-108.

Benedict .S; R.S. Rejitha; and C. Bency Bright. 2012. “Energy Consumption-based Performance Tuning of Software and Applications using Particle Swarm Optimization”. *6th IEEE CSI Int. Conf. on Software Engineering (CONSEG)* 1-6, IEEE.

Brown J.D; J. Demargne; D. J. Seo; and Y. Liu. 2010. “The Ensemble Verification System (EVS): A software tool for verifying ensemble forecasts of hydrometeorological and hydrologic variables at discrete locations”. *Environmental Modelling and Software 25*, 854-872.

Cardona C. M; C. Martin; A. Salerain; A. Castro; D. San Martin; and E. Ayesa. 2011. “CALHIDRA 3.0 – New Software application for river water quality prediction based on RWQM1”. *Environmental Modelling and Software 26*, 973-979.

Chen S. H; A. J. Jakeman; and J. P. Norton. 2008. “Artificial Intelligence techniques: An introduction to their use for modelling environmental systems”. *Mathematics and Computers in Simulation 78*, 379-400.

Coffey .R; E. Cummins; V. O’ Flaherty; and M. Cormican. 2010. “Analysis of the soil and water assessment tool (SWAT) to model Cryptosporidium in surface water sources”. *Biosystems Engineering 106*, 303-314.

Cortes .U; M. Sanchez-Marre; L. Ceccaroni; R. Roda; and M. Poch. 2000. “Artificial Intelligence and Environmental Decision Support Systems”. *Applied Intelligence 13, Netherlands: Kluwer Academic Publishers*, 77-91.

Ge .Y; X. Li; C. Huang; and Z. Nan. 2013. “A Decision Support System for irrigation water allocation along the middle reaches of the Heihe River Basinm, Northwest China”. *Environmental Modelling and Software 47*, 182-192.

Giupponi C. 2007. “Decision Support Systems for implementing the European Water Framework Directive: The MULINO approach”. *Environmental Modelling and Software 22*, 248-258.

Harmel R. D; D. R. Smith; K. W. King; and R. M. Slade. 2009. “Estimating discharge and water quality data uncertainty: A software tool for monitoring and modeling applications”. *Environmental Modelling and Software 24*, 832-842.

Khosrovyan .A; T. A. DelValls; L. Martin-Diaz; C. Morales-Caselles; and I. Riba. 2010. “SQA: a software tool for

integrated sediment quality evaluation based on the Weight-Of-Evidence procedure". *Environmental Modelling and Software 25*, 1483-1484.
Liukkonen M; P. Juntunen; I. Laakso; and Y. Hiltunen. 2013. "A software platform for process monitoring: Applications to water treatment". *Expert Systems with applications 40*, 2631-2639.
Marinoni .O; A. Higgins; S. Hajkowicz; and K. Collins. 2009. "The multiple criteria analysis tool (MCAT): A new software tool to support environmental investment decision making". *Environmental Modelling and Software 24*, 153-164.
McIntyre N. R and H. S. Wheater. 2004. "A tool for risk-based management of surface water quality". *Environmental Modelling and Software 19*, 1131-1140.
Nichita .C and M. Oprea. 2007. "An agent-based model for water quality control". *17th European Symposium on Computer Aided Process Engineering – ESCAPE17*, 1-6.
Sharma .A; M. Naidu; and A. Sargaonkar. 2013. "Development of Computer Automated Decision Support System for Surface Water Quality Assessment". *Computers & Geosciences* , 129-134.
Sulis .A and G. M. Sechi. 2013. "Comparison of generic simulation models for water resource systems". *Environmental Modelling & Software* 40, 214-225.
Sun X.Y; L. T. H. Newham; B. F. W. Croke; and J. P. Norton. 2012. "Three complementary methods for sensitivity analysis of a water quality model". *Environmental Modelling & Software* 37, 19-29.
Tie-zhu .Q and S. Le. 2010. "The Design of Multi-Parameter Online Monitoring System of Water Qulaity Based on GPRS". *The Development of Science and Technology Foundation of Shanxi Province*, IEEE.
Vairavamoorthy .K; J. Yan; H. M. Galgale; and S. D. Gorantiwar. 2007. "IRA-WDS: A GIS-based risk analysis tool for water distribution systems". *Environmental Modelling and Software 22*, 951-965.
Xinyang .Z; Y. Jiaying; S. Jinlin; and X. Zongwu. 2011. "Research of on-line early warning software for source water quality of reclaimed water plant". *Third International Conference on Measuring Technology and Mechatronics Automation*, 1084-1087, IEEE.
Xu H.S; Z.X. Xu; W. Wu; and F.F. Tang. 2012. "Assessment and Spatiotemporal Variation Analysis of Water Quality in the Zhangweinan River Basin, China". *Procedia Environmental Sciences* 13, 1641-1652.
Zhongbo .Y; T. Luxin; C. Hui; B. Bin; and J. Xiaoyan. 2010. "Distributed On-line Water Quality Monitoring System Based on Embedded Web&CAN Bus". *International Conference on Computer Design and Applications (ICCDA 2010)*, V-5, 419-422. IEEE.

AUTHOR BIOGRAPHIES

N.R. WILFRED BLESSING has received B.Tech degree in Information Technology from St. Xavier's Catholic College of Engineering and M.E degree in Computer Science and Engineering from M. N. M. Jain Engineering College which are affiliated to Anna University, Chennai, India. He was awarded his masters degree with Distinction in 2008. He is a research scholar and a researcher in HPCCLoud Research Laboratory, India. He has been researching in Online-based Water Quality Analysis software tools for last 3 years. He has attended many conferences and presented papers. He has also got experience in supporting research projects for creative student community at various organizations.

SHAJULIN BENEDICT graduated in 2001 from Manonmaniam Sundaranar University, India with Distinction. In 2004, he received M.E degree from A.K.C.E, Anna University, Chennai, India. He has completed his Ph.D degree in the area of Grid Computing from Anna University. He was the university second rank holder for his masters. He was affiliated towards the same group and published more papers in International Journals. After his Ph.D award, he joined a research team in Germany to pursue PostDoctorate and successfully awarded. Currently, he is a Professor at SXCCE Research Centre, India and heading the HPCCLoud Research Laboratory which was established by him. He is also a Guest Scientist in TUM, Germany. His research interests include Grid scheduling, Performance Analysis of parallel applications, Cloud Computing, and so forth.

IMPLEMENTATION OF INTELLIGENT COMPUTERIZED SYSTEM MEDICAL DIAGNOSTIC AND TREATMENT FOR DIABETES USING VPN

Bashar S. Bashar
Raaed K. Ibrahim
Foundation of Technical Education
Baghdad –IRAQ
Email: eng_bashar89@yahoo.com

KEYWORKS:

Intelligent agent, VPN software, Diabetes, Diagnostic, Treatment, ID patient

ABSTRACT

This system is designed to improve clinical diabetes, and perform diagnostic and treatment. This application shows how clinics and patient are connected online or offline through application program. In today's life no one has time and money to visit a clinic or hospital. This application will help in getting online or offline diagnostic and treatment with a doctor. Doctor will upload all the patient medical history on the application. This information is visible to only that patient and to the visiting Doctors. Thus, privacy is maintained. As the patient and clinic or hospital is connected online personal information of patient. All this service provided to users at free of cost.

INTRODUCTION

As the importance of the application program is increasing day by day, it is important to manage all the healthcare data online. Now everyone has an internet connection and it is easy to use application. This application will reduce the work of patient as well as doctors. Doctor does not need to take patient's weight, patient's blood group, because all this information is entered at the time of registration of patients in the database. The Doctor will automatically see patient information. There is no more hardware required for patient and doctor. Efficient and effective management of health care is imperative due to the efficient for diagnostic and treatment from a doctor [1]. The program is registering the patient name, gender, ID number, and simple personal and cart Number and accordance with the patient's registration department,. After successful registration, the patients enter to program, if the customer had previously used the reservation registration system; enter the customer's ID number. After the basic situation of the client as well as its previous record of registration and other information is displayed immediately, the patient can be entered directly Into the program; if the client ever had a history of default the system will automatically, In Existing system, the Patient has to pay some money for using that system. But many users prefer using the system to which they don't have to pay money. Every patient got the unique ID at the time of registration. Each doctor in hospital will have a unique ID on the basis of his/her clinic. This ID is useful in the entire process. This ID will be used in searching patient, searching the clinic. It is also useful to maintain privacy of patient and doctor's information. Registered patient data will be verified by the clinic when the patient meets to clinic first time, after that there is no need to verify the data by all the clinics. Clinical data and patient data will be verified by special authority so that there will not be any fraud data. After all this verification Final Unique ID is generated. In purposed of paper is 1- To assist the doctor, 2- It reminds the possible diseases to the doctor on the basis of Symptoms (to overcome human errors like diligence, versatile, tiredness) and 3- Enable a patient to find out the diseases, when no other help is possible.

INTELLIGENT AGENT

Agent Technology has been around for more than two decades, this area of research is still generating as much interest, gave the following definition "an autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future." So an agent is software that has the ability to sense an environment that it is located in, and then carries out some action, based on the information data that it gathers from that environment. The agent is defined [2] see the figure 1.

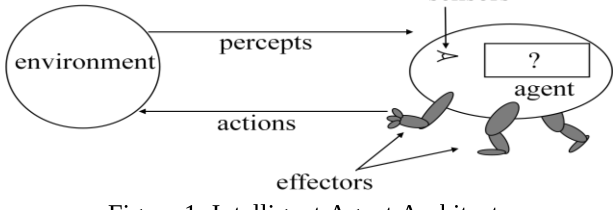


Figure 1: Intelligent Agent Architecture

As the feature of being autonomous, meaning independent or self-directed. So an agent being self-directed should have control over its own actions and not have to rely on the intervention of other agents or even its human creator. Agents also have other features such as being mobile, able to move from one system to another within a networked Environment, another feature is that of intelligence, an intelligent agent is a computer system that is capable of flexible autonomous action in order to meet its design objectives [3]

TYPE 1 AND TYPE 2 DIABETE

Diabetes mellitus is the general name for a group of chronic metabolic diseases characterized by high blood glucose levels that result from defects in insulin secretion. The diabetes is insulin-dependent diabetes mellitus (IDDM) or type 1 [4]. Diabetes was previously known as Non-Insulin Dependent Diabetes Mellitus (NIDDM) or type 2.Type 2 Diabetes can occur in children and adolescents if they are overweight or obese have a family history of type 2 Diabetes and come from a high risk [5].

VPN NETWORK SOFTWARE

A VPN is a private network that uses a public infrastructure (usually the Internet) to connect remote sites or users. The VPN as the name suggest uses "virtual "connections routed through the Interne [6]. There are VPN systems like Hamachi where several security flaws are very apparent. Hamachi uses a centralized server to act as a mediator between two computers in order to create the secure tunnel, the client PC receives an IP address from the Hamachi server, and an IP address for the other PC is necessary and in many cases a user name and password is necessary as well. See the figure 2 [7]

Figure 2 A private networks in Hamachi

SYSTEM IMPLEMENTATION

The system is implemented in multiple clients –server connection, the goal of the system design to offer predictions about patients infected with diabetes, a correct diagnosis and treatment could reduce the risks of diabetes diseases

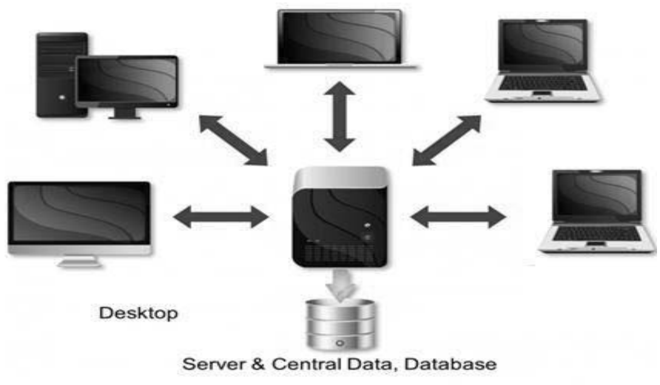


Figure 3: System Architecture

Figure 3 System architecture of the proposed model, the above figure explains working of the proposed system. Doctor, Patient, and Admin are client side users. All these users are connected to the proposed system through the internet. The system is managed by Admin at server side. Proposed system's users are described below

Patient

The patient can enter to program by, doctor's name. If particular doctor was a busy, then the patient can chat with other patients and then scheduling of appointments is done by the program. Secret information about the patient is not visible to other patients; the privacy of the patient is maintained. Time of the hospital and day when the hospital is close is displayed. Emergency contact number of doctors will be displayed

Doctor

The Doctor can see the personal information and allergic information of the patient, and depending upon the allergy, the doctor can give him the prescription. The Doctor can also view the patient history. There is no need of paper to keep a record of the patient. All patient data is stored in the proposed system. It is easy for doctors keep a record of all patients, and also to retrieve the patient information.

Admin

Admin will manage all databases of the proposed system. Admin will update system time to time. And all these updates or delete are available to users without any additional cost.

RELATED PRACTICAL WORK

The System network is principally designed to analysis - making process. Through these registers the patient data in the database, the clinician would be able to enter into this database. The application can be executed in two steps:
1) Logging in for the first time
2) Logging in after the first time below the figure 4

Figure 4: Login Form

The patient have to enter a user name and password when logging in to the application, if the patient is new to the application the patient must register new account by clicking the signup button in the interface. When the patient register a user name that already exist in the database the registration process will be stopped and the application will ask for another user name, in order for the patient to perform the registration process, the patient must own a card that holds a 16 digits, this card is provided by the doctor only If a patient enters all the information correct the data will be saved in database in SQL server and save the IP address of computer and the program will work as shown the figure 5

Figure 5: Patient Interface

The patients send information such as question and lab analysis hormones, and Immunity, biochemistry & hematology and bacteriology & serology, data of diabetes…etc. The patient answer the questions that is show in the interface of patient for one time only and send to the database as shown the figure 6

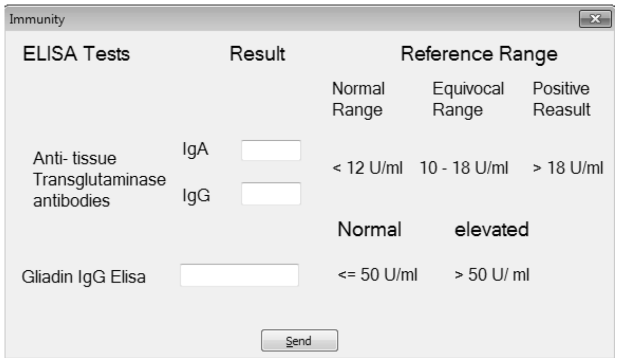


Figure 6: Screen for the Immunity lab

When the Patient completes the questioner and send the lab results, the doctor can now press the diagnostic button in figure 7, where a potential treatment and disease options will be available to the doctor and the doctor can choose the best treatment that he sees best. Shown the figure 8 and 9

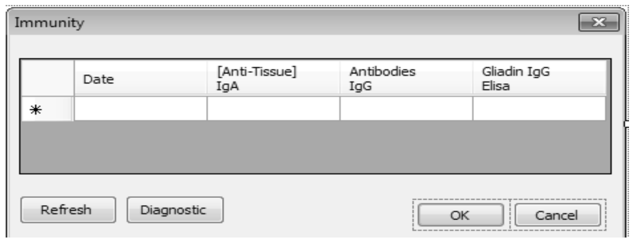


Figure 7: immunity lab at doctor interface

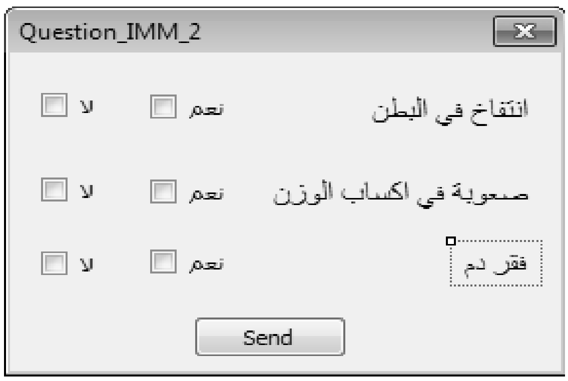


Figure 8: Screen Question of the patient

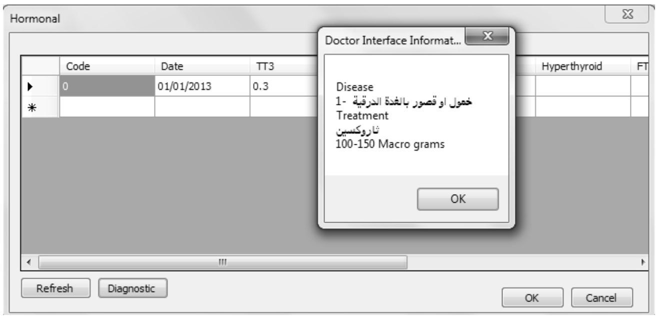


Figure 9: Screen the Diagnostic result

When patient have any problem in connection between doctor and patient such as server shutdown or network error the communication is disconnected, a massage will appear to the patient shown in figure 10. The patient can press the “Yes “button to resend the signal between the server and client, if server doesn’t exist the patient can select the offline status shown the figure 11

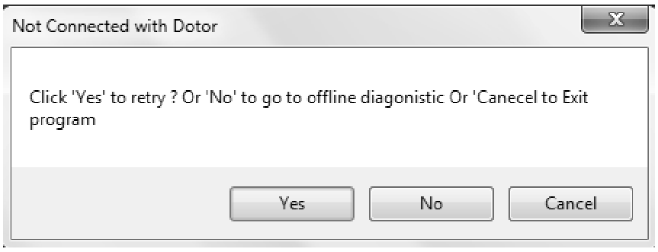


Figure 10: Disconnect with the doctor

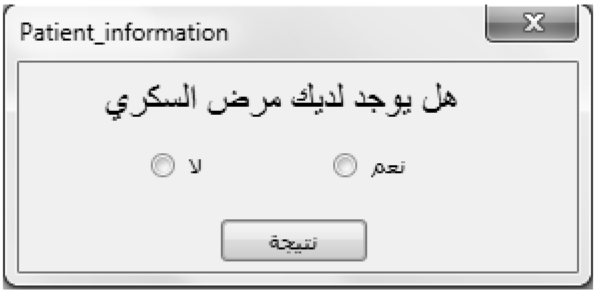


Figure 11: Screen for offline

CONCLUSION

The Proposed system is used to connect all clinics in a city to a hospital. The system is available to doctors, patient. The system assigns unique ID patient. The system manages all records. Also ensures security, integrity, privacy and availability of data. Management patient health software allows patients to maintain health.

REFERENCES

Hang Bo, 2010 “Web based long-distance appointment registered System CCTAE, International Conference, vol. 3, Page(s): 232 – 235

S. Franklin and A. Graesser. Is it an Agent, or just a Program: A Taxonomy for Autonomous Agents. [Online]. Available: http://www.msci.memphis.edu/~franklin/AgentProg. html

N. R. Jennings and M. Wooldridge. (1998). Applications of Intelligent Agents. [Online]. Available: http://agents.umbc.edu/ Introduction/jennings98.pdf

Crawford JM. Liver, biliary tract, and pancreas. 1991 In: Robbins SL, Cotran RS, Kumar V Pocket companion to Robbins pathologic Basis of disease Philadelphia, PA: WB Saunders; p. 313-7.

P. Harris, L. Mann, and 2012 “diabetes management in general practice” An electronic version of these guidelines is available at www.diabetesaustralia.com.au, ISBN: 978 1875690 190

www.hamachi.com-about.com

AUTHOR BIOGRAPHIES

B.Sami Bashar has received Engineering computer Tech. Degree from Al-mamon University College and Msc degree in Engineering computer Tech. he is Waiting discussion search for Master degree after two month from now. He has been researching in GUI interface and Database Program in last 2 years.

Dr. Raaed K. Ibrahim graduated in 2000 from Al-Nahrain University. In 2002, he received M.E degree from Al-Nahrain University; He has completed his PhD degree from Al-Anhrain University in 2008. He was affiliated towards the same group and published more papers in many countries such as Germany and Italy. Currently, he is a assistant Professor at college of electrical and electronic

Modelling Virtual Graduation Projects Coordinator: From Object Oriented to Agent Oriented

Al-Hanouf Al-Dayel[1*], Ashwag Al-Swayah[2*], Buthina Al-Dhafian[3*]

Dr. Ghada Al-Hudhud[4#], Dr. Abir Benabid Najjar[5*]

[*] *Software Engineering Department, King Saud University*
Riyadh, Saudi Arabia

[1] a.aldayel0@gmail.com

[2] ashwag-ksu@hotmail.com

[3] b.r.al.dhafian@gmail.com

[5] abbenabid@ksu.edu.sa

[#] *Information Technology Department, King Saud University*
Riyadh, Saudi Arabia

[4]galhudhud@ksu.edu.sa

KEYWORDS
Agent-oriented methodologies, Software engineering, Multi-agent systems, switching an object oriented software to agent oriented software

ABSTRACT

Nowadays, agent-oriented software development technologies have evolved rapidly; it is emerging as a new paradigm for constructing intelligent more autonomous software systems. Therefore, several methodologies have been proposed. As a result, it is difficult to determine the most appropriate methodology for specific project within different domains; especially when re-engineering current object oriented software system. Thus, the objective of this paper is to present a comparison and evaluation of five well-known agent-oriented methodologies based on an attribute based framework which in order to decide which methodology better suites evolving an object oriented software system into agent oriented software system. The case study is going to be rebuilding a virtual graduation project coordinator in Information Technology Department at King Saud University.

INTRODUCTION

Due to the difficulty involved in the design and development of complex software systems, wide ranges of software engineering paradigms have been developed (e.g. object-oriented programming, structured programming, procedural programming and declarative programming) [1]. In recent years, Agent-oriented software engineering (AOSE) is emerging as a new paradigm for constructing software systems where agents are exploited to build complex applications for industrial software system. AOSE seems to be a potentially powerful paradigm and is thought off one of the most interesting research fields in the software engineering community since they are able to handle the structure and behaviour of complex real world applications. Agent technology is applied to many domains including Electronic-Commerce (ecommerce) applications, robotics, network security, and computer games, virtual organization and management applications e-learning systems [2]. Hence AOSE has been introduced as one of the ultimate solutions in future software development environments. Meanwhile, another interesting implementation of the agent oriented software engineering is rehabilitating the object oriented software engineered systems.

The work presented in this paper explains the life cycle of moving an object oriented old graduation project coordinator system into an agent oriented virtual coordinator of the graduation projects. The authors of the paper decided to evaluate the agent's development methodologies in order to select the most appropriate one for object evolution. Hence, evaluating the AOSE methodologies' strengths and weaknesses offers an overview analysis of these methodologies that can help software developers to find out which one is the most appropriate for rehabilitating and evolving the current used systems in a certain domain. In this paper, the authors evaluate five different agent-oriented methodologies based on a framework which addresses four major aspects: concepts and properties, notations and modelling techniques, process, and pragmatics. These methodologies are: GAIA [9-14], MaSE [15-20], MESSAGE [21-23], Prometheus [24-29] and Tropos [30-34]. These methodologies were selected based on (i) the familiarity of the agent community with them, (ii) the availability of their descriptive documentations and (iii) The long time period in which these methodologies had been developed based on users feedback. Moreover, the selected AOSE methodologies cover all the three Agent-oriented categories; methodologies which are specially tailored to the AOSE (GAIA), methodologies that directly extend or adapt object-oriented methodologies (MaSE, MESSAGE, Prometheus), and methodologies that adapt knowledge engineering models or other techniques (Tropos).

This paper is organised as follows. Section 2 introduces background of object oriented software engineering and compares it with agent-oriented approaches. In section 3, we briefly introduce five AOSE methodologies. Section 4 introduces a framework for comparing and evaluating AOSE methodologies, and then we apply it on the selected methodologies. Section 5 presents the case study for the rehabilitating an object oriented software into agent oriented software according to the selected methodology.

I. BACKGROUND

At present, Agent-Oriented Software Engineering (AOSE) has high level of expansion; According to the needs of complex systems that interact with their users. There are similar aspects between the OOSE and the AOSE, as well as there are significant difference aspects between them [3, 4]. Main component in agent-oriented approach is the agent while it is the object in object-oriented approach. Agent can be treated as an active object, but has a mental state unlike object in OOSE [5]. According to [6], the agent can be described as an encapsulated software component that owns several characteristics as flexibility and autonomous ability to achieve their goals.

How objects or agent communicate with each other is the main difference between them. Objects and agents rely on messages to communicate with each others, but the essence of the difference is: objects use method invocation to pass these massages between them, while agents discriminate different message's types and use knotted types and knotted protocols to negotiate [7]. Agent is distinct from object by analyse messages that pass between agents. Also, it can determine if the action is executed in this time or not [7].

Since four decades ago, Object-Oriented has developed several methodologies to avoid software crisis and organize the system development process [8]. There are three views of the system in object-oriented methodologies: (i) *Static view* that describes structure objects and their relationships,
(ii) *Dynamic view* that shows the interactions between objects, and (iii) *Functional view* that shows the data flow of the objects' methods [7]. Based on [4], the agent-oriented methodologies have two viewpoints: (i) *The external viewpoint;* its basis depends on the division of the system. In addition, it involves agent model and interaction model. These models respectively describe agents with their interaction and the other clarifies the relationships between agents in the system and relationship between these agents and external systems, and (ii) *The internal viewpoint* is about describing each agent internally. This viewpoint has three models developed in this stage: the belief, the goal and the plan model. The belief model shows how the agent works on its beliefs about the environment. The goal model shows goals and events that agent performs them or complies. The plan model describes how the agent achieves its goals.

A. GAIA

GAIA (Generic Architecture for Information Availability) is a general methodology that has been developed by Wooldridge et al. for agent-oriented analysis and design of agent systems in 2000. Then, it was extended to support open multi-agent system (MAS). The extension is based on the key consideration that an organization is more than a collection of roles (as it was considered in Gaia v.1), and that additional organizational abstractions must be identified (Zambonelli et al., 2003) [9].

GAIA supports both levels of micro (agent structure) and macro (agent society) of the development in agent systems. It has two main phases: the analysis phase and design phase. It starts with the analysis phase, which deals with collecting the functions needed to understand MAS. And then, the design phase which aims to identify the organizational structure for the MAS in efficient and reliable way, and to complete accordingly the preliminary roles and interactions models. An overview of the methodology and its models is shown in Fig 1.

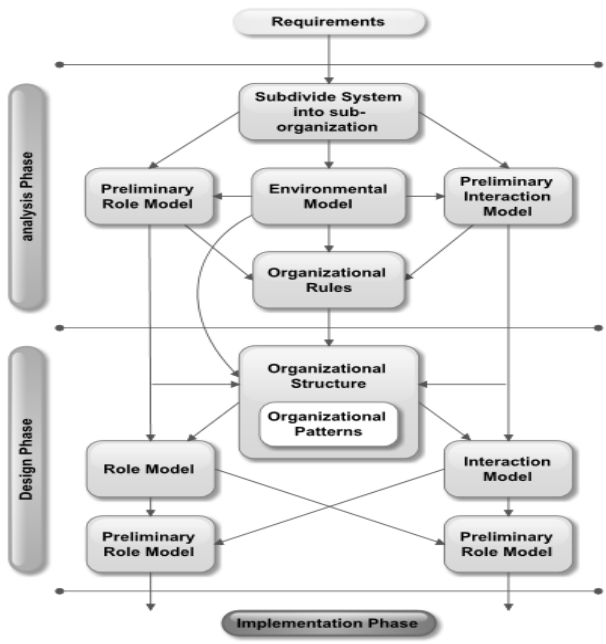


Figure 1 GAIA Methodology Models

B. MaSE

The Multi-agent System Software Engineering (MaSE) is a general purpose methodology for analysing, designing and developing heterogeneous multi-agent systems (DeLoach et al., 2001). It supports the whole software development life cycle and it can lead a system developer from an initial system specification through implementation using a set of system models [15]. MaSE is independent of any particular agent architecture, multi-agent system architecture, programming language, or communication framework [16].

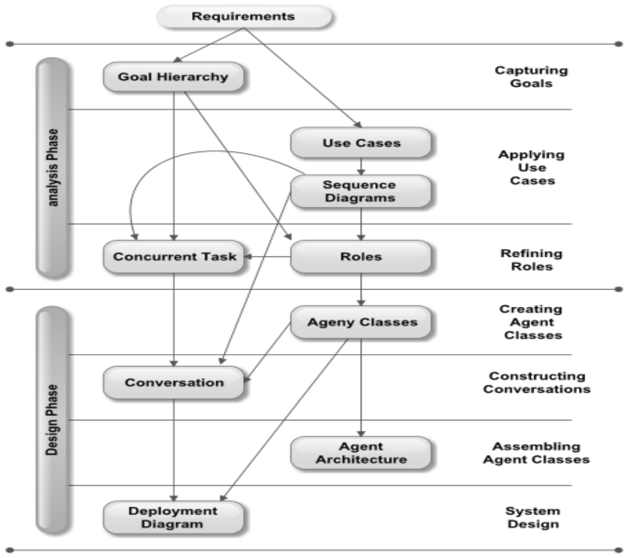


Figure 2 MaSE methodology models

The general operation of MaSE follows the series steps shown in Fig 2, with outputs from one step that become inputs to the next step. This methodology is iterative across all phases with the purpose that all the details for models will be added later which can emphasize the ability to track changes throughout the process [17]. The development process in MaSE consists of seven steps, divided into two phases: analysis and design. Fig 2 illustrates the process of MaSE methodology.

The Analysis phase produces a set of roles whose tasks have to include the overall system requirements.

1. Capturing goals identifies the main system goals from the initial system specification and structures them into a Goal Hierarchy Diagram by importance [15].
2. Applying use-cases captures use cases from the initial system requirements and restructures them as a Sequence Diagram to help determine the actual communications required within a multi-agent system [17].
3. Refining roles ensures all the system goals are accounted by mapping the goals of the Goal Hierarchy Diagram into roles. Roles can define agent's classes and capture system goals during the design phase [17]. Once the roles have been identified, the system must define how a role accomplishes its goals by defining detailed tasks and assigned them to specific roles [15]. Roles and their associated tasks are captured and represented in a Role Model.

The Design phase transforms the roles and tasks into agent types and conversations in order to define the overall system organization. In this phase, there are four steps included; creating Agent classes, constructing conversations, assembling Agent classes and system design.

1. Creating Agent Classes: identifies agent classes from component roles and presented them in an Agent Class Diagram; the product of this phase which can depict the overall system organization consisting of agent classes and the conversations between them [17].
2. Constructing Conversations: defines a coordination protocol between two agents, or in other meaning, it defines the details of the constructed conversations between two agents. A conversation consists of two Communication Class Diagrams: initiator and responder [16].
3. Assembling Agent Classes: creates agent class internals. This step is performed in parallel with the preceding step (constructing conversations). This is achieved via two steps: defining the agent architecture and defining the components for the selected architecture [18]. Designers can have the choice to either design a new architecture from scratch or use a predefined architecture such as Belief-Desire-Intention (BDI), planning, reactive, knowledge based, and a user-defined architecture [19]. Additionally, a designer can develop new components in the architecture or use predefined components.
4. System Design: takes the agent classes and instantiates them as actual agents, like instantiation objects from object classes in object-oriented programming [17]. The defining of system parameters (numbers, types and locations of agents within a system) are represented in Deployment Diagrams.

Table 1 summarizes all the models for each step in MaSE methodology.

TABLE 1: MASE PHASES

Phases	Models
1- Analysis Phase	
a. Capturing goals	Goal Hierarchy
b. Appling use cases	Use Cases, Sequence diagrams
c. Refining Roles	Concurrent Tasks, Role Model
2- Design Phase	
a. Creating Agent Classes	Agent Class Diagrams
b. Constructing Conversations	Conversational Diagrams
c. Assembling Agent Classes	Agent Architecture Diagrams
d. System Design	Deployment Diagrams

MaSE is supported by a CASE tool (called agent Tool). AgentTool supports the analysis and design activities in each step of MaSE phases and it has the ability to validate the various models, generate automatic transformation of models, and generate skeleton code [20].

C. MESSAGE

MESSAGE methodology started its activities on June 1999 and finished in June 2001, found by EURESCOM [21]. MESSAGE conceder as an extended methodology from object oriented methodologies [21]. It passes through the analysis and design phases of multi-agent system (MAS). MESSAGE mostly relies on UML notations and concept. Although these notations and concepts undergone some modifications to suit the concept of agent. This extension of UML with new modifications called AgentUML (AUML). AUML has been adopted within MESSAGE. MESSAGE and UML have almost the same meta-modeling language; and Meta Object Facility (MOF), too [22].

AgentUML is an extension of UML (Unified Modeling language). Definitely, this modeling language takes into consideration the variation between objects and agents. Agent has the power to initiate action without external invocation and it can to refuse or modify an external request) [23]. Fig 3 shows some updated notations used in AUML.

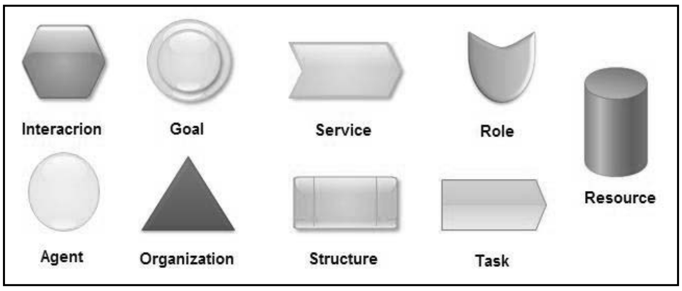


Fig. 3 The updated notations of AgentUML

Additionally, MESSAGE picks some concepts from the higher level artificial intelligence (AI) which called the knowledge level [22].

TABLE II: COMPARISON BETWEEN GAIA AND MAS-COMMONKADS

Methodology Name	Analysis Models	Design Models	Defects
GAIA	2	3	- Represent subset of analysis models - Design models not clearly explained and no detailed design.
MAS-CommonKads	6	3	- Does not have unifying semantic framework and notations.

Agent/Role view highlights on agent and role individually. Schemata used for each Agent/Role. And it supported by diagrams to its characteristics. This view determines each goal is responsible for what, what resources it controls...etc. Interaction view delves deeper into the agent/role by showing the interaction between them. Domain view shows concepts of the domain [22, 22].

D. Prometheus

The Prometheus is an agent-oriented methodology. It was designed to be a practical methodology which can be used by industry practitioners and undergraduate students in summer training [24]. Prometheus is suitable to non-expert; it tough to and used by undergraduate students [25]. The distinguished feature in this methodology is detailed design phase which makes the transition to the implementation phase easier than other methodologies (i.e. GAIA, Tropos) [26]. Prometheus includes all software development phase. So, these detailed and inclusive of all phases made this methodology used to develop intelligent agent systems [27].

In Prometheus methodology, the system passes through three phases: system specification, architectural design and detailed design [25]. In accordance with [24], Prometheus provides dynamic models, structural models, and descriptor in each phase. The dynamics models focus in the behaviour of the system (i.e. Interaction diagrams). The structural models describe the structure of the system as well as the structure of its components (i.e. Goals diagrams). The descriptor is textual forms which detailed entities in details (i.e. Functionalities). Prometheus's phases can be described as follow:

In Prometheus, there are scenarios that can describe actors and their interaction with the system. Scenarios can be considered as use cases in Object-Oriented [28]. Scenarios illustrate the system operations and functionalities. Functionalities obtained by grouping goals into sub-groups of goals that have the same purpose [24]. These functionalities apply modularity principle (one of software engineering principles). The descriptor used to describe the system functionalities [24]. Each scenario described using a name, description and a triggering event [24]. One of the system specification aspects is to identify the system goals and sub-goals [26].

First set of goals captured by defining a goal for each use case (Scenario) [26]. In this phase, it is important to identify clearly the interface between the agent system and its environment (i.e. hardware, software); because of the agent system definitely it will effect on its environment [24]. As well as it effected by this environment [24]. So they need interface between them which organize these interactions.

E. Tropos

Tropos is an agent-oriented software engineering methodology which offers a modelling language on the basis of a multi-agent paradigm. It proposed by (J. Mylopoulos, M. Kolp and P. Giorgini, 2000) [30].

Tropos is based on two fundamental notions: (i) the concept of agent and all relation mentalistic concepts (such as goals, plans) that used in all phases of software development, (ii) it contains early phases of requirements analysis which can help to understand the system environment during the software work, and it the type of interactions between software and human agents [31]. Tropos aims to support all activities of the analysis and design from the very early phases of requirements engineering down to implementation. It applied all phases of the software development process: Early Requirements, Late Requirements, Architectural Design, Detailed Design and Implementation [30].

According to [31, 32], the basic concepts that will be used during each phases of development process are: (i) Actor which describes an entity that has main objectives, achieve inside the system, and represents the physical, social or software agent as well as role (is an abstract properties of the behaviour of a social actor) or position (represent a various roles), (ii) Goal which describes strategic interests for actor, (iii) Plan which describes a way to do something at level of abstraction. It implemented to satisfying a goal or softgoal, (iv) Resource which describes physical and the entity's information, (v) Dependency which describes dependency between two actors. One actor depends on the other actor to achieve the desired goals, to implement a specific plan or to provide a resource. The first actor called depender, while the other called dependee. The object (goal, plan or resource) which is located in the centre of dependency called dependum, (vi) Capability which describes the actor ability to identify, select and implement a plan to achieve its goal, and (vii) Belief which describes the knowledge of actor.

There is a set of modelling activities that constructed using the concepts of Tropos and used during the development phase. Specially, in early requirements phase then will be refined in the other phases. These models are: actor modelling, goal modelling, plan modelling, and capability modelling. These models detailed below based on [31, 32]:

Actor modeling aims to identify and analyze actors of the environment, actors in the system and agents. Dependency modeling describes actors which dependent to each other. Actor modelling and dependency modelling will be represented using actor diagrams (Fig 4)., which

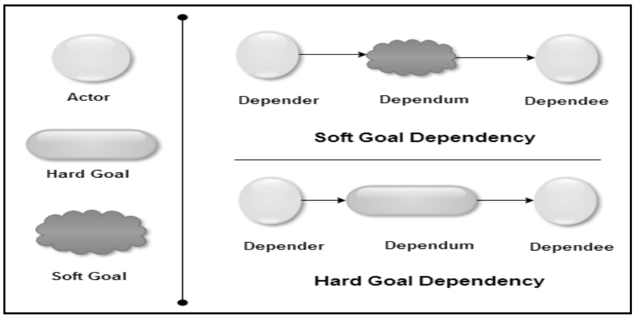


Fig. 4 Symbols of plan diagram

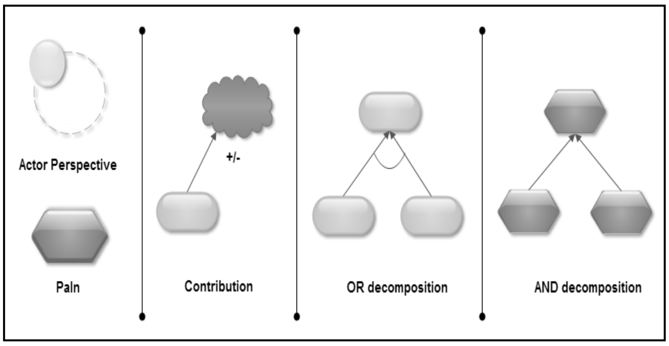


Fig.5 Symbols of goal diagram

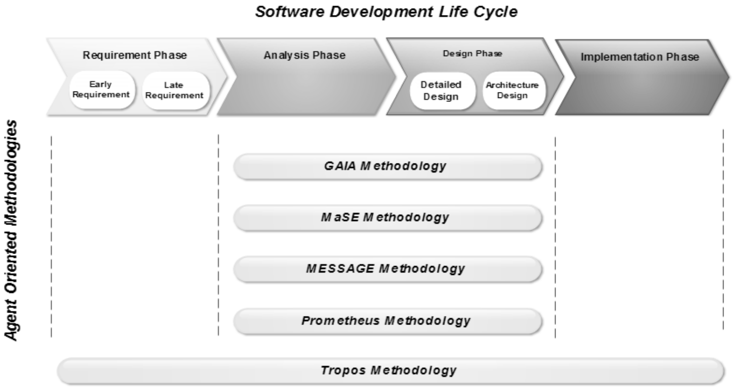


Fig.6 Comparing AOSE methodologies in the development life -cycle coverage

contribute positively or negatively in the fulfillment of the goal to be analysed and/or (iii) decomposition that combines AND/OR decompositions of a root goal into sub-goals, and accurately models a goal structure. Plan modelling is supplement goal modelling that can be considered as an analysis technique. Goal modelling and plan modelling are represented using goal diagram (Fig 5). Capability modelling aims to define, choose, and execute a plan. Capability modelling is represented using capability and plan diagrams.

II. THE COMPARISON BETWEEN AGENT-ORIENTED METHODOLOGIES

The challenge of selecting an appropriate methodology for designing a Multi-Agent system is by understanding and evaluating the difference between them (strengths, weaknesses).

In this paper, our evaluation consists of a set of criteria which addresses both software engineering processes and agent-oriented properties.

Table III COMPARISON OF AGENT ORIENTED METHODOLOGIES

Concept	GAIA	MaSE	MESS-AGE	Prome-theus	Trop os
I. Concepts and Properties Comparison					
Autonomy	Y	Y	Y	Y	Y
Mental Mechanism	N	Goal, task	Goal, task	Goal	Goal, soft-goal, task
Adaptation	Y	N	No details	No details	Y
Concurrency	Y	Y	No details	Weak	Y
Communication	No details	Y	Y	Y	Y
Collaboration	Y	Y	Y	Y	Y
Agent Abstraction	Roles	Roles	Goals/ tasks	Functions	Social actor s
Agent-oriented	Y	N	N	N	N
II. Notations and modeling techniques Comparison					
Expressiveness	Y	Y	Y	Y	Y
Complexity	Role	Goal, role, refine-ment	No details	Agent type	Decom pos-ition of goals, tasks
Modularity	Y	N	Y	Y	Y
Executable	N	N	Y	No details	N
Refinement	N	Y	Y	Y	Y
Traceability	Y	Y	No details	Y	Y
III. Process Comparison					
Specification	Role analysis	Use cases goal and role analysis	No details	Scenarios and Goals analysis	Stakeh older analysi s
Life-cycle Coverage	Y	Y	Y (RUP)	Y	Y
Architecture Design	N	Y	Y	Y	Y
Implementation Tools	N	Y	N	Y	Y
Deployment	Y	Y	No details	N	N
IV. Pragmatics Comparison					
Tools Available	N	Y	Y	Y	N
Required Expertise	N	N	No details	N	N
Modelling Suitability	N	N	No details	No details	BDI
Domain Applicability	Y	Y	Y	No details	Y
Scalability	Y	Y	Y	Y	Y

The adopted framework [2] is similar to Sturm & Shehory framework in their divisions [35]. It covers four major aspects of each AOSE methodology: Concepts, Modeling language, Process and Pragmatics. Table 3 summarises the main aspects and criteria considered in the evaluation framework that support for autonomy, communication and collaboration.

– In GAIA, autonomy is expressed by the fact that the role encapsulates its functionality. While it is expressed by system role in MaSE.
– In Prometheus, autonomy is expressed by the fact that the functionalities are encapsulated in goal.
– Only GAIA and Tropos are able to adjust their activities in dynamically changing environments (support the adaption).
– GAIA supports communication by its own interaction protocols and the communication paths between agents. However, the communication architecture and the content of a message are not expressed in any GAIA models. MaSE also supports communication by its own protocol analyzer, and the communication paths between agent classes in constructing conversations step.
- MESSAGE supports communication between agents. During analysis or design phases, the interactions between entities will have been created. The protocol of interaction (i.e. FIPA interaction) needs to be addressed early in analysis phase; because if it was addressed in the designs phase, there will have to be a significant effort made to recast interactions [21]. The interaction among agent illustrated in interaction view in the analysis phase.

III. Case Study: Modelling Virtual Coordinator for Graduation Project

In order to smoothly move from the object oriented system developed system into agent oriented, the graduation project coordination system at the department of information technology at King Saud University is re-engineered into agents.
Selecting the AOSE methodologies that directly extend or adapt object-oriented methodologies among the currently known methodologies in the filed such as MaSE, MESSAGE, Since MaSE is based on building the mental mechanism of the agent to include goals and task and connect the goals to agents roes in the system. The authors decided to select this methodology to extend their old object oriented system into agent oriented system. Next step was to identify the goals and tasks for each agent according to the agent's role in the system as an organization that communicates with the other organization (another agent).
Each agent is considered as an organization that interacts with another agent in order to provide it with the required task. The collection of agents in the evolved system:

1. Scheduler Agent
2. Course Assessment Report Manager; CAR Agent
3. Calendar Agent
4. Assignment Manager Agent
5. Groups Manager Agent

Below is presented the description of each agent's goal and roles

Assignment Manager Agent

Goal: To automatically produce the assignment process for all groups to supervisors

Role:

1. Get the supervisors list from the proposals accounts
2. Get the students groups ID
3. Get the groups average GPA
4. Get the students preferences lists
5. Run the assignment algorithm

Constrains:

1. Each group should only be assigned one project
2. The project should do not be assigned twice

CAR Agent

Goal: To automatically produce the final assessment report for one course

Role:

1. Get the weekly evaluation grads from each supervisor account for each project
2. Get the weekly evaluation grads from each supervisor account and from committees members for each project
3. Get the students learning outcomes indirect analysis from the students surveys
4. Get the students learning outcomes direct analysis from the committees grades and supervisors grades

Constrains:

1. Each grad submitted by the committee member should be submitted after the presentation date, and does not exceed 40; i.e. out of 40.
2. Each weekly grade submitted by the supervisor should be submitted before the final presentation date.

Scheduler Agent

Goal: To automatically produce the final presentations schedule

Role:

1. Get the excluded time slots/ days
2. Decide the presentations slots during these days
3. Adopt all the preferences of the students, supervisors and committees
4. Produce the final presentations schedule fo rall groups, committees.

Constrains:

1. Each committee should have all the projects to be examined by this committee at the same day.
2. Each supervisor; if a committee member, will have the projects supervised by this supervisor at other day

IV. CONCLUSION

The need for a clear strategy and supported AOSE methodology increases in order to develop complex systems. Unsuitable choice of methodology can lead to have many risks and problems.

Overall, this paper briefly presented the restructuring redesigning of the old object oriented graduation project coordinator into agents. IN order to restructure the object oriented developed systems; the authors presented a comparison between the different agent oriented methodologies in order to select the suitable one that extends the object oriented designs. The main concepts of agent-oriented as a basis to understand the methodologies. Mainly, five agent-oriented methodologies are presented. These methodologies may include as extensions to Knowledge engineering methodologies (i.e. Tropos) or to Object-oriented methodologies (i.e. MaSE, MESSAGE and Prometheus). Additionally, GAIA methodology appears as a pure agent-oriented methodology. Sufficiently, This paper describes each methodology from their software development stages perspective. Finally, the evaluation framework between these methodologies is applied. It covers four major aspects of each AOSE methodology: Concepts, Modelling language, Process and Pragmatics. Since AOSE methodologies are currently evolving. And many of AOSE methodologies have recently developed. Choosing the best suited to particular system can be considered as an issue that requires understanding and comparing between all AOSE methodologies. In future research, we will expand a number of different well-known methodologies and compare between them. Therefore, we will seek to apply an Agent methodology to convert an object-oriented based system to become Agent-oriented system. And this can help to expand our research in AOSE area.

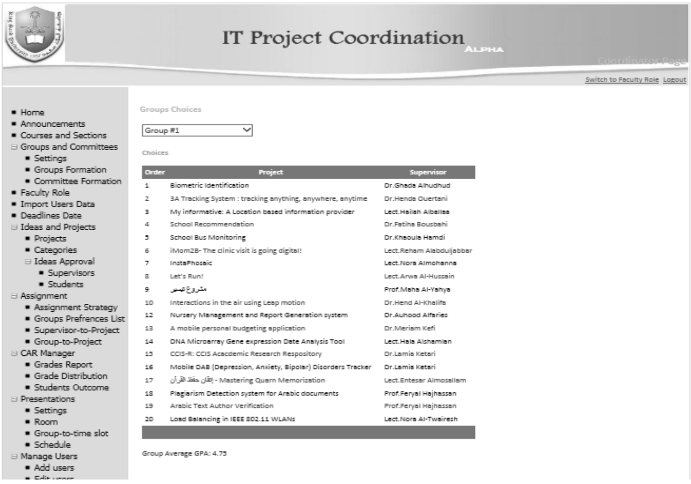


Fig 8: All agent roles are represented into separated units in the system interface to the left

REFERENCES

[1] N. R. Jennings and M. Wooldridge, "Agent-Oriented Software Engineering", In Handbook of Agent Technology (ed. J. Bradshaw) AAAI/MIT Press, 2000.

[2] C. Lin, K. M. Kavi, F. T. Sheldon, and T. E. Potok, “A Methodology to Evaluate Agent Oriented Software Engineering Techniques,” International Journal of Multiagent and Grid Systems, pp. 1–20, 2005.

[3] B. Burmeister, “Models and methodology for agent-oriented analysis and design,” Working Notes of the KI, vol. 96, no. 96–06, p. 52, 1996.

[4] D. Kinny, M. Georgeff, and A. Rao, "A methodology and modelling technique for systems of BDI agents." Agents breaking away: Springer Berlin Heidelberg, Proceedings of the Seventh European Workshop on Modelling Autonomous Agents in a Multi-Agent World MAAMAW'96, (LNAI Volume 1038). Springer-Verlag: Heidelberg, Germany, 1996.
[5] Y. Shoham, "Agent-oriented programming," Artificial intelligence, vol. 60, no. 1, pp. 51–92, 1993.
[6] M. Wooldridge, "Agent-based software engineering." IEE Proceedings-software 144.1, 1997: 26-37.
[7] C. A. Iglesias, M. Garijo, and J. C. González, "A survey of agent-oriented methodologies," framework, vol. 2, p. 34, 1999
[8] U. A. Khan, I. Al-Bidewi, and K. Gupta, "Object-Oriented Software Methodologies: Roadmap to the Future," International Journal of Computer Science, vol. 8.
[9] L. Cernuzzi, T. Juan, L. Sterling, and F. Zambonelli, The Gaia Methodology: Basic Concepts and Extensions. In Methodologies and Software Engineering for Agent Systems". Kluwer Academic Publisher, 2004.
[10] A. Pnueli, "Specification and development of reactive systems," in Information Processing 86. Elsevier Science Publishers B.V.: Amsterdam, The Netherlands, 1986.
[11] N. Genza and E. Mighele, "Review on Multi-Agent Oriented Software Engineering Implementation," International Journal of Computer and Information Technology, vol. 02, no. 03, pp. 511–520, 2013.
[12] T. Abdelaziz, "Towards a Comprehensive Agent-Oriented Software Engineering Methodology," Doctoral Dissertation, Universität Duisburg-Essen, Germany, 2008.
[13] J. Gonzalez-palacios and M. Luck, "Extending Gaia with Agent Design and Iterative Development," in Proceedings of the 8th international
[14] F. Zambonelli, N. R. Jennings, and M. Wooldridge, "Developing Multiagent Systems□: The Gaia Methodology," ACM Transactions on Software Engineering and Methodology, vol. 12, no. 3, pp. 317–370, 2003.
[15] S. A. Deloach, E. T. Matson, and Y. Li, "Exploiting Agent Oriented Software Engineering In Cooperative Robotics Search And Rescue," International Journal of Pattern Recognition and Artificial Intelligence, vol. 17, no. 5, pp. 817–835, 2003.
[16] S. A. Deloach and M. Wood, "Developing Multiagent Systems with agentTool," in Proceedings of the 7th International Workshop on Intelligent Agents VII. Agent Theories Architectures and Languages, 2001, pp. 46–60.
[17] M. F. Wood and S. A. Deloach, "An Overview of the Multiagent Systems Engineering Methodology," in Proceedings of the 1st International Workshop on Agent-Oriented Software Engineering, 2001, vol. 1957, pp. 207–221.
[18] A. S. Rathore and D. Roy, "Agent Oriented Software Engineering Methodologies: A Review," in Proceedings of the 2ed international conference on computer applications, 2012, pp. 136–147.
[19] D. J. Robinson, "A Component Based Approach to Agent Specification," Master thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson Air Force Base Ohio, USA, 2000.
[20] S. A. Deloach, M. F. Wood, and C. H. Sparkman, "Multiagent Systems Engineering," International Journal of Software Engineering and Knowledge Engineering, vol. 11, no. 3, pp. 231–258, 2001.
[21] R. Evans et al., "MESSAGE: Methodology for engineering systems of software agents," EURESCOM, EDIN, pp. 0223–0907, 2001.
[22] G. Caire et al., "Agent oriented analysis using MESSAGE/UML," in Agent-oriented software engineering II, Springer, 2002, pp. 119–135.
[23] B. Bauer, J. P. Müller, and J. Odell, "Agent UML: A formalism for specifying multiagent software systems," International journal of software engineering and knowledge engineering, vol. 11, no. 03, pp. 207–230, 2001.
[24] M. Winikoff and L. Padgham, "The prometheus methodology," Methodologies and Software Engineering For Agent Systems, pp. 217–234, 2004.
[25] K. H. Dam and M. Winikoff, "Comparing agent-oriented methodologies," in Agent-Oriented Information Systems, 2004, pp. 78–93.
[26] L. Padgham, M. Winikoff, and D. Poutakidis, "Adding debugging support to the Prometheus methodology," Engineering Applications of Artificial Intelligence, vol. 18, no. 2, pp. 173–190, 2005.
[27] L. Padgham and M. Winikoff, "Prometheus: A methodology for developing intelligent agents," in Agent-oriented software engineering III, Springer, 2003, pp. 174–185.
[28] K. H. Dam, M. Winikoff, and L. Padgham, "An agent-oriented approach to change propagation in software evolution," in Software Engineering Conference, 2006. Australian, 2006, p. 10–pp.
[29] L. Padgham, J. Thangarajah, and M. Winikoff, "Prometheus Design Tool.," in AAAI, vol. 8, 2008, pp. 1882–1883.
[30] J. Brinkkemper and A. Solvberg, "Tropos: A framework for requirements-driven software development," Information Systems Engineering: State of the Art and Research Themes, p. 11, 2000.
[31] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos and A. Perini, "Tropos: An Agent-Oriented Software Development Methodology," Autonomous Agents and Multi-Agent Systems, vol. 8 ,no.03, pp.203–236, 2004.
[32] F. Giunchiglia, J. Mylopoulos, and A. Perini, "The tropos software development methodology: processes, models and diagrams," in *Agent-Oriented Software Engineering III*, Springer, pp. 162–173, 2003.
[33] A. Perini, P. Bresciani, P. Giorgini, F. Giunchiglia, and J. Mylopoulos, "Towards an Agent Oriented Approach to Software Engineering.," in *WOA*, vol. 2001, pp. 74–79, 2001.

[34] Agent Oriented Software, ‘‘JACK™ Intelligent Agents Agent Practicals’’, Agent Oriented Software Pty Ltd, 2012. [Online]. Available: http://www.aosgrp.com. [Accessed: Nov. 27, 2013].

[35] A. Sturm and O. Shehory, “A Framework for Evaluating Agent-Oriented Methodologies,” in Fifth International Bi-Conference Workshop on Agent-Oriented Information System, no. 1, pp. 60–67, 2003.

ELECTRONICS SIMULATION

A COMPARATIVE STUDY BETWEEN FAT TREE AND MESH NETWORK-ON-CHIP INTERCONNECTION ARCHITECTURES

Azeddien M. Sllame
Computer Network Department
Faculty of Information Technology
University of Tripoli
Aziz239@yahoo.com

Amani Hasan Abdelkader
Computer Science Department
Faculty of Science
University of Tripoli
Tripoli, Libya

KEYWORDS

Network-on-Chip, Fat tree, Mesh, Simulation

ABSTRACT

In this paper we do a comparative study between mesh and fat-tree based Network-on-Chip (NOC) systems. To do the comparison we used two tools; the *gpNoCsim* and fat-tree simulators. The analysis shows both the efficiency and the applicability of mesh and fat tree structures to NOC targeting system on chip designs. Even though the mesh structure regularity made it suitable for physical implementation, the observed results strongly convinced us to say that the scalability and higher bandwidth of the fat tree structure will make it the preferred interconnection architecture for future massively parallel NOC systems.

INTRODUCTION

System-on-Chip (SOC) is a model that deals with building a system using billions of transistors on a single silicon die (chip). SOC may contain many modules that have variety of signals including digital, analog, mixed-signal, and it often includes radio frequency (RF) functions, buses, memory elements, image processing blocks (e.g. MPEG core), digital signal processing (DSP) cores, general-purpose processors (CPU) and configurable logic blocks (CLB) of field programmable gate arrays (FPGA) and does it all on a single chip as seen in figure 1. Such pre-designed functional blocks (modules, components) are commonly called "cores" intellectual property (IP cores). Nowadays, mobile phones, cable and satellite TV *set-top-boxes*, and portable media devices are typical examples of SOC systems.

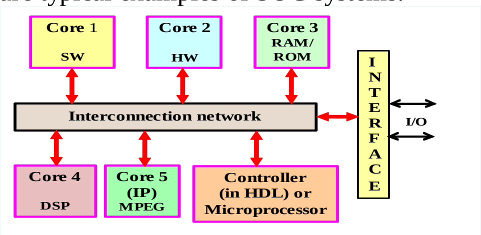


Figure 1: Generic SOC Structure

The most fundamental distinguishing characteristic of a SOC is its structure and connectivity complexity. In practice, most of SOCs are *multiprocessor systems-on-chips* (MPSOCs) because it is too difficult to design a complex SOC without making use of multiple CPUs, various DSP cores, and numerous memory pieces, RF modules. The interconnect topology in the MPSOC model in the last two decades was either point-to-point or bus communication links. Nowadays, instead of connecting the top-level SOC cores by using buses or routing dedicated wires, they are connected to an *interconnection network* that routes packets between them; see figures 2; which now named as "*Network-On-Chip*". NOC is a communication subsystem on an integrated circuit (commonly called a "chip"), typically made between IP cores composing of SOC. NOC technology applies networking theory and methods to on-chip communication and brings outstanding improvements over conventional bus and point-to-point interconnections; i.e. "routes packets, not wires" (Dally et al. 2004).

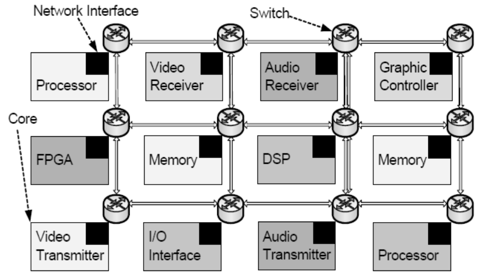


Figure 2: SOC Based on Mesh NOC Interconnection

More precisely, in NOC model, when one IP core is idle, other IP blocks continue to make use of the network resources. Hence, NOC improves the scalability of SOCs, and the power efficiency of complex SOCs compared to other designs. This approach has the benefits of being modular and scalable, well-structured, reusable components, and flexible with higher bandwidth. NOC approach has efficient performance that can be adapted to different workload needs, while maintaining the generality of application development methods and practices. On the other hand, the bus approach has the performance degradation as the number of cores goes beyond 16 cores; and the point-to-point wiring suffers from power dissipation, cross talk delays due to routing inside the chip. Besides that, interconnection networks have been already used in many super-computers for more than five decades, so it is have been systematically researched, thoroughly verified and investigated, and well-documented. Butterfly fat tree (BFT) and mesh architectures are typical examples of those interconnection networks, see figure 2 and (Duato et al. 2002; Dally et al. 2004). Interconnection networks can be classified according to different characteristics. Their topologies fall into two classes *static* (or direct) and *dynamic* (or indirect). In a static network, point-to-point links interconnect the network nodes (IP cores) in some fixed topology; a regular topology as mesh or a hypercube is common examples of this category. A dynamic network allows the interconnection pattern among the network nodes to be varied dynamically: this is accomplished by some form of switching. Examples of dynamic networks include fat trees and multistage interconnection networks. When we speak about networking; we need: *switching*, *routing*, and *flow control* of the packets of the transferred message across that network. The switching technique defines how messages are propagated through the network. More precisely, the switching mechanism defines the hardware and software protocols for transmitting and buffering data when sending a message between neighboring switches (routers) (Duato et al. 2002; Dally et al. 2004). A variety of switching techniques have been proposed to support communication across the multiple network channels between the source and destination. The most commonly used techniques are; *circuit switching, store-and-forward routing, virtual cut-through, wormhole routing*; more information can be found in (Duato et al. 2002; Dally et al. 2004). The packet can be defined as the smallest unit of communication containing routing information (e.g. destination address) and sequencing information in its header. Its size is of order of hundreds or thousands of bytes or words. The packet is composed of group of data units named as *flits;* header flit and many data flits. The flit is the smallest unit of information at the link layer; its size is one of several bytes. Flits can be several types and flit exchange protocol typically requires several cycles to transfer one single flit. *Routing* determines the path that will be selected to transfer the packet(s) to reach its destination; it must be decided within each intermediate router which output channel(s) that will be selected to forward incoming packets to their final destinations. *Flow control* defines the synchronization protocol between sender and receiver nodes and determines right procedures to be taken in case of full buffers, busy output channels, faults, deadlocks, etc. In this paper, we are using NOC simulators that allow us to model NOC systems by specifying both the behavior of the network nodes and the way of the communication switching and routing protocols of the whole network.

This paper is organized as follows: interconnection networks are briefly described in section 2. Fat tree network structure is illustrated in section 3. Mesh networks is outlined in section 4. Section 5 outlines theoretical comparison between mesh and fat tree networks. Finally, practical simulation results are given in section 6.

INTERCONNECTION NETWORKS

Interconnection networks have been developed to realize fast and reliable communication systems in parallel systems in last five decades. This fact emphasizes the importance of interconnection networks to overall parallel system performance since any parallel system that employs more than one processor per application program must be designed to allow its processors to communicate efficiently.

FAT TREE ARCHITECTURE CONCEPTS

The fat tree is a type of interconnection network, where the processing elements (IP cores; for simplicity) are interconnected by a tree structure, in which the IP cores are at the leaves of the tree, and the interior nodes are switches. An advantage of a tree structure is that communication distances are short for local communication patterns. Moreover, the fat tree is a tree structure with redundant interconnections on its branches; the number of interconnections increases as the root is reached (Leiserson 1985). The purpose is to increase the bandwidth at higher levels, where it is most needed. Because it is not feasible to provide a channel between every pair of nodes, the network channels are shared among the IP nodes. Messages are used to communicate between sending and receiving nodes, which means construction of paths that are consist some intermediate switches (for switching/routing purposes) along the specified paths from the sources to the destinations. Figure 3 shows a butterfly fat tree with 64 IP blocks (cores) interconnected by suitable number of switches in intermediate levels. The IP nodes are placed at leaves in zero level and switches are placed in higher levels. We can calculate number of levels by the relation:

$L = \log_4 N$, Where N is the number of IP nodes.

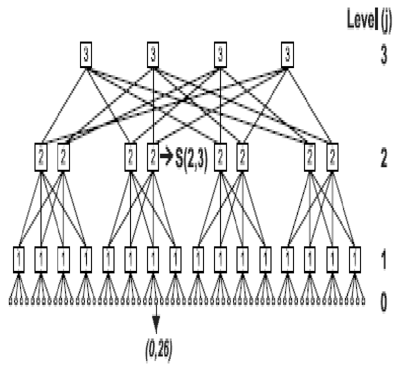


Figure 3: Butterfly Fat Tree Structure with 64 IP Cores

In the network illustrated in figure 3 we have 3 levels, and the switches are placed in levels ranging from $l > 0$ and $l >=$ L. Each IP node is denoted by pair (i,0) where i is ranging from (0-63) which denotes the index of the IP node in the level zero, each IP node has two ports to connect with its parent switches, each port has two unidirectional physical links. Each level of the network has the number $\frac{1}{2^{(l-1)}} \times \frac{N}{4}$ of the switches and the total number of switches in the network is the summation of the number of switches in each level. Each switch is represented by a pair of coordinates (i, l), where i represents the index of the switch in the level and l represents the level of the switch, the pair (5,2) represents switch no. 5 in the level no. 2. Each IP node at the coordinate *(i, 0)* has the parent at coordinate (*p,1*) and *p=i/4*. For example if we have the IP node (0, 62), it has the parent switch (15, 1). Each switch has two parent (p) coordinates (*p1,l+1*) and (*p2,l+1*)

$$p1 = \frac{i}{2^{(l+1)}} \times 2^{l} + i \bmod 2^{(2-l)}$$

$$p2 = p1 + 2^{(l-1)}$$

For example, if we have the switch (15,1) , then from the above relations it has the parents: *P1*=6 and *p2*=7. And each switch has four children (switches or nodes). The *least common ancestor algorithm* is commonly used with fat tree designs as an adaptive routing algorithm; see (Duato et al. 2002). In addition, wormhole switching is accompanied with virtual channel mechanism is usually used as a switching technique in recent fat tree designs (Pande et al. 2003).

MESH ARCHITECTURE CONCEPTS

Mesh is another type of interconnection networks. 1-D meshes are made from linear arrays of processing elements and incrementally scalable. The most practical meshes are, of course, 2-D and 3-D ones (Bononi et al. 2007). In a mesh network, the nodes are arranged in a k dimensional lattice of width *w*, giving a total of *wk* nodes; usually *k*=1 (linear array) or *k*=2 (2D array). Communication is allowed only between neighboring nodes. All interior nodes are connected to *2k* other nodes. The major advantage of the mesh is its simplicity. All links are short and balanced and the overall layout is very regular. The routers are low radix with up to *C+4* input and output ports. The major disadvantage is the large number of hops that flits have to potentially go through to reach their final destination (proportional to "N" for N routers). XY is a dimension order routing which is a typical minimal turn deterministic algorithm that is commonly used with mesh designs (Duato et al. 2002; Dally et al. 2004). The algorithm determines to what direction packets are routed during every stage of the routing, which routes packets first in x- or horizontal direction to the correct column and then in y- or vertical direction to the receiver. XY routing is working well on both mesh and torus topology. Addresses of the routers are their xy-coordinates, as seen in figure 4.

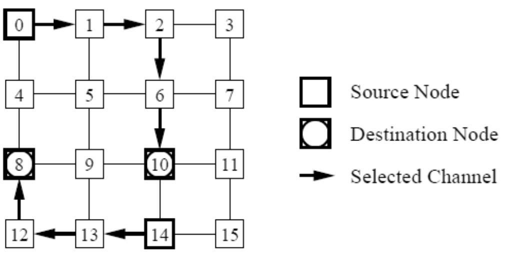


Figure 4: Mesh Interconnection Networks

THEORETICAL COMPARISON BETWEEN FAT TREE AND MESH NOC

In this section we aim to provide detailed comparison between mesh and fat tree as NOC architectures. The data provided here are scattered in many papers with different point of views, e.g. (Wingard 2001; Guerrier et al. 2000; Pande et al. 2003; AlTawil et al. 1997; Salminen et al. 2007; Jantsch et al. 2003; Bononi et al. 2007; Dally et al. 2001; Kumar et al. 2002). To the best of our knowledge it is the only study that collected all of such useful information in one document. More details found in (Amani 2013).

Topology:

- Mesh networks belongs to direct type interconnection networks; where point-to-point interconnects the network nodes in fixed regular topology.
- Fat tree is the typical example of the indirect (dynamic) interconnection network; which allows changing of the interconnection arrangement among the network nodes dynamically through the use of network's switches.
- In the 2D mesh topology there are as many switches as IP cores and the design of the switches is always the same (i.e. cost overhead).
- In fat tree topology the number of switches is independent on the number of processing elements and switches with different designs can be used. Packet has many routes to destination due to structure connectivity layout.
- Mesh structure has connections between 4 neighbor nodes
- For mesh topology it is simple to compute the distance between current and destination node as the sum of the offsets in all the dimensions.

Out of order reception of packets :

- Adaptive topology in fat tree allows out-of- order reception of the packets per message flow, which adds complexity to the implementation.
- Deterministic routing in 2D mesh has in-order packet delivery which makes it simple to implement.

Network traffic balance :

- In mesh networks deterministic routing performs well under uniform traffic only.
- In fat tree adaptive routing makes network traffic balance better than deterministic routing thus allowing obtaining higher throughput to the network and low latency.

Deadlock, livelock, starvation:

- Mesh uses XY deterministic routing which is considered as deadlock and livelock free.
- Fat tree designs employ adaptive routing in which there is a possibility of livelock and starvation. Hence, a special care should be taken during switch design process in order to avoid deadlock, livelock and starvation.

Routing:

- Mesh uses XY deterministic routing, in which all packets follow the same route between given source-destination pairs. However, it always chooses the shortest path with in-order flow control.
- In the traditional XY routing, the traffic does not extend regularly over the whole network because the algorithm causes the biggest load in the middle of the network.
- Deterministic routing provides low routing latency and good reliability when the network is not congested.
- Fat tree uses adaptive routing, in which the route can vary from packet to packet depending on the network situation. And it can adapt to network congestion conditions (can do re-routing, out-of-order transmission).

Fault tolerance:

- Mesh structure has low fault tolerance against the possible faults in the network, since it is fixed connected topology.
- Adaptive routing in fat tree increases the fault tolerance. However. Fault tolerance can be achieved via error

detection and correction, adaptive routing, but require special attention to avoid deadlocks.

Congestion control:

- Mesh networks can't dynamically respond to network congestion, which will lead to throughput and network efficiency degradation.
- Fat tree can avoid network congestion using adaptive routing, of course by using more information about the network situation. This in turn, increases the complexity of implementation in terms of area and cost.

Latency and throughput:

- In mesh networks the latency and throughput increases with increase in mesh size.
- Adaptive routing is more complex in implementation and provides higher throughput and low latency.

Network utilization:

- XY deterministic routing in mesh networks has under-utilization of the network resources.
- XY deterministic routing tends to send packets toward the center of the mesh when the contention is high. That is in turn will lead to early network saturation and performance degradation.
- XY routing performs well under the uniform distribution traffic.
- Fat tree designs employ adaptive routing in which rerouting and out-of-order enhances network utilization.

Scalability:

- Scalability can be defined as the property which exhibits performance proportional to the number of IP cores employed in the network structure. Moreover, bandwidth (BW) of the network is considered as performance metric.
- Fat tree is more scalable as its BW is the highest as compared to mesh. In addition, fat tree is recursively scalable design.
- The BW in mesh designs is fixed by the number of resources, i.e. it does not scale.
- The BW in fat tree networks depends on the switch design.

Energy dissipation:

- Energy dissipation increases linearly with the increase of the number of virtual channels in switch design for both mesh and fat tree.
- In mesh networks one switch in every network node increases its cost and power consumption.

Physical realization:

- Mesh networks are particularly easily mapped to physical space with uniformly short wires. The simplest case is when the network is a mesh with the same number of dimensions as the physical dimensions of the packaging technology.
- It is not possible to map fat tree topology directly into two dimensions provided by a silicon chip as in the case of mesh topology, without increasing the length of some interconnection wires proportionally to the number of cores. This will decrease the clock frequency dramatically and interference the performance.

PRACTICAL RESULTS

The simulator

General Purpose Simulator for Network-on-Chip Architectures simulator (gpNoCsim) has been described in (Hemayet et al. 2007) is used in obtaining practical comparison results. In addition, the fat tree simulator that has been developed under the supervision of the first author, described in (Sllame et al. 2012), is used to validate some results. *gpNoCsim* is an open-source tool developed in Java, component based simulation framework for NOC architectures. Version 1.0 of gpNoCsim contains the implementation of mesh, torus, butterfly fat tree, extended butterfly fat tree networks. Hence, it supports doing comparison of performance parameters among different networks such as throughput, latency (average packet delay), link utilization, buffer utilization, average hop count, average packet not produced. gpNoCsim uses the wormhole switching technique supported with virtual channels in the input and output ports. Figure 5 shows the general structure of the IP cores (nodes) and switch communication flow details. The node defines the necessary methods that are used for messaging and traffic configuration. Messages are generated from a source node and forwarded to its parent switch. From that switch, after appropriate routing and arbitration, the message either forwarded to an adjacent switch or passed to the destination node.

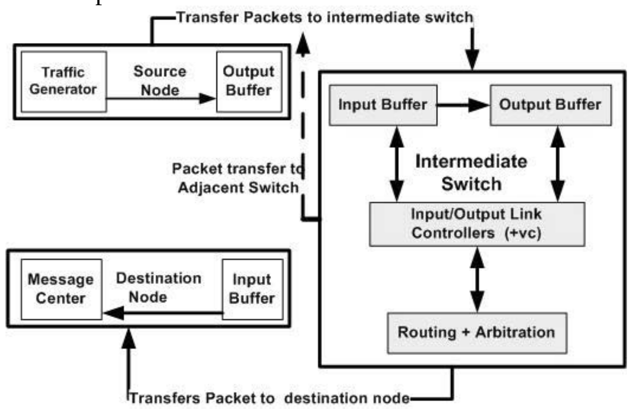


Figure 5: IP Core/Switch Internal Structure and Packet Communication Flow

Nodes are connected to the parent switches through one input physical link and one output physical link. Each node has its address, its generated message list to hold the generated messages, and received message list to hold the received messages from different nodes. Each node has *traffic generator* method which is responsible for packets generation with fixed (or random) lengths, producing random data and sending it to random destinations (it like a seed to feed the network with packets for the simulation be approaching real systems). This method also divides the packet into one header flit and more data flits and one tale flit to indicate the end of the packet (Hemayet et al. 2007; Bononi et al. 2007).

The analysis

We have analyzed the networks (mesh, fat tree) using the simulator with the following input parameters: IP nodes sizes (16, 32, 64) with different number of virtual Channel / link: (2,4,6,8,10) and with different number of flits / buffer : (2,4,8,10) with average packet length of [200,300,400] bytes, and number of simulation cycles [1000, 2000, 3000].

Mesh NOC uses XY deterministic routing algorithm, Fat tree uses least common ancestor adaptive routing algorithm. Wormhole switching technique supported with virtual channels mechanism is used in both mesh and fat tree architectures. The following are the main performance parameters in the simulator:

a) *AVG INTER ARRIVAL*: The mean rate of message generation (in simulation cycles) in the resource nodes. Actual inter packet generation interval is calculated by exponential distribution using this parameter. By changing this parameter network load can be increased or decreased.

b) *AVG MESSAGE LENGTH* : The mean length of message (i.e. packet length, in bytes) generated in the IP nodes. By changing this parameter network load can be increased or decreased also.

c) *FLIT LENGTH*: Length of the flit (both header and data flit) in bits. If the size is too small to hold the minimum information to carry header flit then it is increased in number of bits.

d) *VC COUNT* : The number of virtual channels in each of the physical channels in the network, e.g. 2,4,6,8, 10.

e) *FLIT PER BUFFER* : Size of the input buffer and output buffer in flit unit, i.e. fixed amount of buffer is incorporated in input/output buffer of the network.

Table 1 shows some results about the effect of buffering of mesh and fat tree NOC networks when IP cores =16. The results shows a clear increase in average packet delay for mesh as the number of flits/buffer increases, while the increase is not clear in the fat tree. Moreover, in fat tree the number of packets leaving switches is higher than that of mesh.

Table 1: Effect of flits buffering for IP cores =16

Mesh			
Number of flits / Buffer	Throughput	Throughput [Flits leaving Switch]	Avg Packet Delay
2	0.1605	0.54325	45.957446808
4	0.2135	0.781312	65.427350427
8	0.185812	0.66375	56.030927835
fat tree			
2	0.16525	1.1795	43.915254237
4	0.171625	1.185333	43.33620689
8	0.164812	1.120666	50.234234234

Figure 6 shows the results of comparing mesh and fat tree in terms of throughput and number of virtual channels per one physical link. The number of flits/buffer is constant=2. Both designs have nearly linear throughput when virtual channels changes from 2 till 6. But, we see that in the fat tree case the throughput started decreasing when the number of virtual channels reaches 8 per physical channel due to switching overhead, while mesh network drops when virtual channels =8 and increases again. Figure 7 illustrates the relationship between average throughput per switch [Flits leaving Switch] and number of virtual channel for 16 IP cores networks with the number of flits/buffer =2. AS shown in the figure 7, the fat tree has higher throughput per switch which comes from adaptive routing used within the fat tree switch. Figure 8 clarifies the relationship between average packet delays (ns) with average message length (bytes) for 64 IP cores networks. This figure produced when the number of virtual channels =4 for each physical link and number of flits/buffer=4. The figure shows that fat tree has higher average packet delay than mesh networks when the packet length= 300,400 bytes. Figure 9 describes the relation between average packet delay and number of virtual channnels for 64 IP cores. We used here constant number of flits/buffer =2 and the change were in number of virtual channel. As shown in the figure 9 the fat tree has higher average packet delay than mesh and it goes down in a linear fashion as the number of virtual channels increase. But the mesh behaved differently in different values. The relation between number of VC/link and number of flits stored in buffers for 64 nodes is illustrated in figure 10. The figure shows buffer size (2 to 8) flits per buffer for fat tree NOC. For every virtual channel no./link there are 3 situations for buffers. As the number of buffers increase the average packet delay decrease in the network for most of the cases. On the other hand figure 11 illustrates the same parameters for mesh networks. However, as the number of buffers increase the average packet delay has not changed too much for most of the cases for mesh networks.

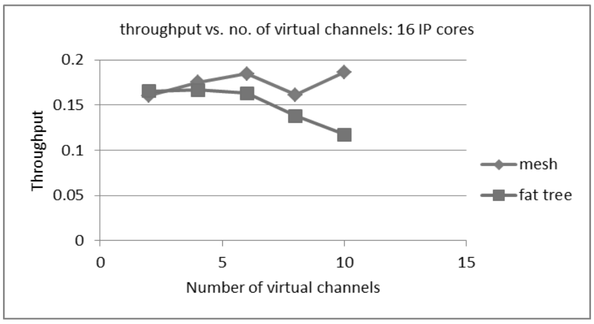


Figure 6: Throughput and Number of Virtual Channels for 16 IP Cores

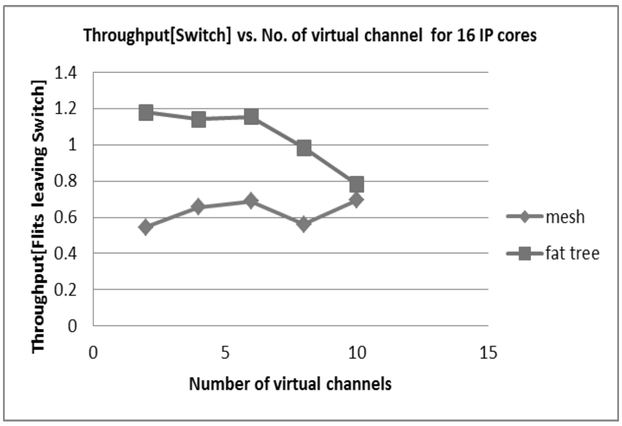


Figure 7: Average Throughput per Switch (Flits Leaving Switch) and Number of Virtual Channel for 16 IP Cores

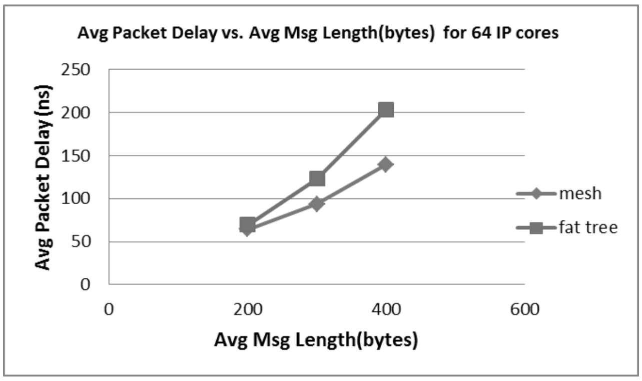


Figure 8: Average Packet Delays (ns) with Average Message Length (bytes) for 64 IP Cores

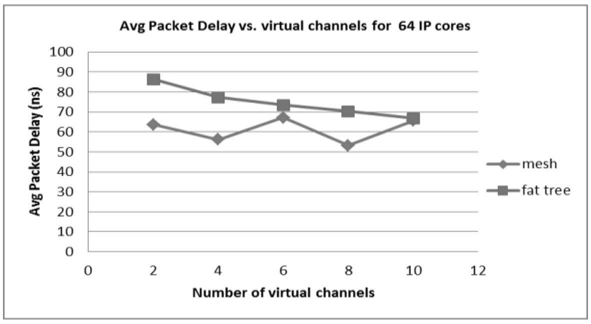


Figure 9: The Relation Between Average Packet Delay and Number of Virtual Channnels for 64 IP Cores

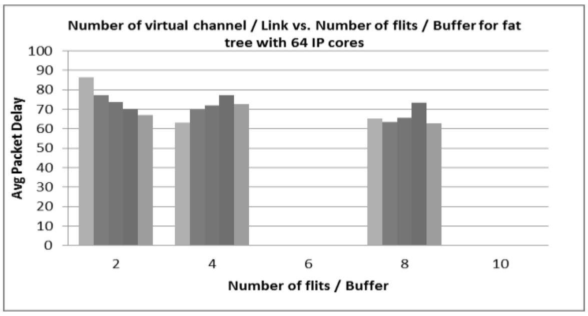


Figure 10:. Different Values of Buffer Size with Different Virtual Channels for Fat Tree with 64 IP Cores

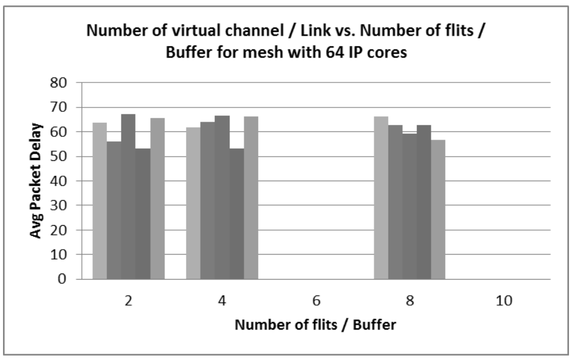


Figure 11: Different Values of Buffer Size with Different Virtual Channels for Mesh with 64 IP Cores

CONCLUSION

The main goal of this paper was to analyze the 2D-mesh and fat-tree architectures as a NOC interconnection networks candidates. The evaluation process has been done using available open-source simulators. The comparison includes: *routing, switching methods used in the switches, effect of buffering, effect of virtual channel technique*, *effect of packet length*. We believe that the scalability and higher bandwidth of the fat tree network makes it the preferred NOC for future massively parallel NOC systems. Even though some papers reported that the most common topology that now mostly used in NOC design is 2D mesh due to structure regularity which helps physical mapping (60% of NOC cases: Jantsch et al. 2003).

REFERENCES

Duato J., S.Yalamanchili, L. Ni. 2002. *Interconnection Networks– An Engineering Approach*. Morgan Kaufmann Publishers, San Francisco, USA.

Duato, J., O. Lysne, R. Pang, and T. M. Pinkston. 2005. "Part I: A theory for deadlock-free dynamic reconfiguration of interconnection networks." *IEEE Trans. on Parallel and Distributed Systems* 16, No. 5 (May), 412–427.

Dally, W. J., and B. Towles. 2004. *Principles and Practices of Interconnection Networks*. Morgan Kaufmann Publishers, San Francisco, USA.

Dally W. J. and B. Towles. 2001. "Route packets, not wires: On-chip interconnection networks." In *Design Automation Conference*, pp. 684–689.

Pande P. P., C. Grecu, A. Ivanov, R. Saleh. 2003. "Design of a Switch for Network-on-Chip Applications." In *Proceedings of ISCAS*, Bangkok, Vol. V, pp. 217-220.

Kumar S., et al. 2002. "A Network on Chip Architecture and Design Methodology." In *Proceedings of ISVLSI*, 117-124.

Wingard D. 2001. "Micro Network-Based Integration for SoCs." In *Proceedings of DAC2001*, LasVegas, Nevada, USA.

Guerrier P., A. Greiner. 2000.**"**A generic architecture for on-chip packet switched interconnections." In *Proceedings of DATE Conference and Exhibition*, pp. 250 –256.

Grecu C., P. P. Pande, A. Ivanov, R. Saleh. 2004. "A Scalable Communication-Centric SoC Interconnect Architecture." In *IEEE International Symposium on Quality Electronic Design ISQED*, San Jose, California, USA, 22-24 (March).

Leiserson C. 1985. "Fat-Tree: Universal Network for Hardware-Efficient Supercomputing." *IEEE Trans. On Computers* C-34, No. 10 (Oct.), 892-901.

Sllame A. M., A. Elasar. 2012. "An Efficient Switch for Fat Tree Network-on-Chip Interconnection Architecture" In *IEEE International Conference on Computer Systems and Industrial Informatics (ICCSII'12)*, Sharjah, UAE.

Bononi L., N. Concer, M. Grammatikakis, Coppola, M. R. Locatelli. 2007. "NoC Topologies Exploration Based on Mapping and Simulation Models." In *Proceedings of the 10th Conference on Digital System Design Architectures, Methods and Tools*. IEEE, CA.

Hemayet H. M., A. Ahmed, T. Islam AI-ayeemn and A. Md Mostofa. 2007. "GPNOCSIM - A General Purpose Simulator for Netwotk-on-Chip." In *Proceedings of the International Conference on Information and Communication Technology IEEE ICICT07*.

Dale K. P. 2004. "Modeling and simulation verification and validation challenges." *Johns Hopkins APL Technical Digest* 25, No. 2, 163-172.

Al-Tawil K. M., M. Abd-Elbarr, F. Ashraf. 1997. "A survey and comparison of wormhole routing techniques in a mesh networks." *IEEE Network* 11, 38–45.

Salminen E., A. Kulmala, and T. Hamalainen. 2007. "On network-on-chip comparison." In Euromicro DSD, 503–510 (Aug).

Jantsch A, H. Tenhunen. 2003. *Networks on Chip*. Kluwer Academic Publishers, USA.

Amani H. A. 2013. **"**A Comparative Study between Mesh and Fat-Tree Network-On-Chip Systems," MSc thesis, University of Tripoli, Libya.

BIOGRAPHY

AZEDDIEN M. SLLAME (BSc, MSc, PhD): was born in Benghshir, Libya. He received his B.Sc in Computer Engineering in 1990 from Engineering Academy, Tajoura, Libya. He earned his M.Sc in Computer Science and Technology from Brno Technical University, Czech Republic in 1997. In 2003 he got his PhD in Information Technology from Brno University of Technology. He published more than 25 scientific papers in many international conferences and journals in the areas of designing of high-performance digital systems, system-on-chip, network-on-chip designs and evolvable hardware systems. He is now assistant professor with faculty of information technology (University of Tripoli, Libya). Email: aziz239@yahoo.com.

VHDL PROTOTYPING OF A 5-STAGES PIPELINED RISC PROCESSOR FOR EDUCATIONAL PURPOSES

Safaa S. Omran and Hadeel Sh. Mahmood
Computer Engineering Techniques
College of Electrical and Electronic Techniques
Baghdad, Iraq
E-mail: omran_safaa@ymail.com , hadeel_shakir@yahoo.com

KEYWORDS
VHDL, Pipeline, MIPS, Data Hazard, Control Hazard, FPGA.

ABSTRACT

This paper describes the **VHDL** (**V**ery High Speed IC **H**ardware **D**escription **L**anguage) implementation of a complete 5-stages, 32-bit, pipelined **MIPS** (**M**icroprocessor without **I**nterlocked **P**ipeline **S**tages) processor with integer multiplication and division support. The processor design supports all the 50 integer instructions including 25 R-type, 9 J-type and 16 I-type instructions. At the beginning, the processor's complete design is divided into three units: the pipelined datapath unit, the pipelined control unit and the hazard unit which solves all the problems of data hazards and control hazards. Then a program that finds the factorial of number 6 is executed and results are discussed. The **VHDL** design of the complete pipelined **MIPS** processor is implemented by using (*Xilinx ISE Design Suite 13.4*) program and configured on *Xilinx Spartan-3AN* **FPGA** (**F**ield **P**rogrammable **G**ate **A**rray) starter kit.

INTRODUCTION

In a single-cycle MIPS processor, an entire instruction can be executed in one cycle and the cycle time is limited by the slowest instruction (Omran and Mahmood, 2013).
Pipelining is an implementation technique in which multiple instructions are overlapped in execution (Hennessy and Patterson, 2012). Therefore, pipelining is achieved by subdividing the single-cycle MIPS processor into several parts. Each part is called a stage. Such that, in the 5-stages pipelined MIPS, five instructions can be executed simultaneously, one in each stage (Linder and Schmid, 2007). Each stage takes a single clock cycle, so under ideal conditions, the speed up from pipelining is approximately equal to the number of pipeline stages (Hennessy and Patterson, 2012).
The VHDL design of a basic 5-stages, 32-bit, pipelined MIPS processor has been made by several previous researches which implement the simple design that can execute the basic instructions (*add*, *sub*, *lw*, *sw*, *and*, *or*, *beq*, *bne* and *j*) (Robio, 2004. Singh and Parmar, 2012).
Since one of the major utilities of VHDL is that it allows the synthesis of a circuit or system in a programmable device or in an ASIC (**A**pplication **S**pecific **I**ntegrated **C**ircuit) (Pedroni, 2004. Perry, 2002), the MIPS-32 compatible CPU (**C**entral **P**rocessing **U**nit) was designed, tested, and synthesized using VHDL.

INSTRUCTION SET FOR PIPELINING

Designing instruction set for pipelining required:
First, fetching instructions in the first pipeline stage and decoding them in the second stage required that all MIPS instructions to be of the same length.
Second, reading the register file in the second stage at the same time that the hardware is determining what type of instruction was fetched required that MIPS to have only a few instruction formats, with the source register fields being located in the same place in each instruction.
Third, calculating the memory address in the execute stage and then accessed in the following stage required that memory operands only appear in loads or stores in MIPS.
Fourth, operands must be aligned in memory (Hennessy and Patterson, 2012).

PIPELINE HAZARDS

Pipelined processor may suffer from hazards. Hazards occur when the next instruction cannot execute in the following clock cycle. There are three different types of hazards:

1. **Structural hazard:** occurs when the hardware cannot support the combination of instructions that required to execute in the same clock cycle [Alekya. 2011].
2. **Data hazard:** occurs when an instruction tries to read a register that has not yet been written back by a previous instruction. Data hazards are solved with forwarding (bypassing) and stalls.
3. **Control hazard:** occurs when the decision of what instruction to fetch next has not been made by the time the fetch takes place. Control hazards are solved with either stalling until the branch is complete which is too slow or branch prediction (static or dynamic) and flushes, or delaying slot.

IMPLEMENTATION

Figure 1 shows the complete design of the pipelined MIPS processor which consists of:

A. **Pipelined datapath unit.**
B. **Pipelined control unit.**
C. **Hazard unit.**

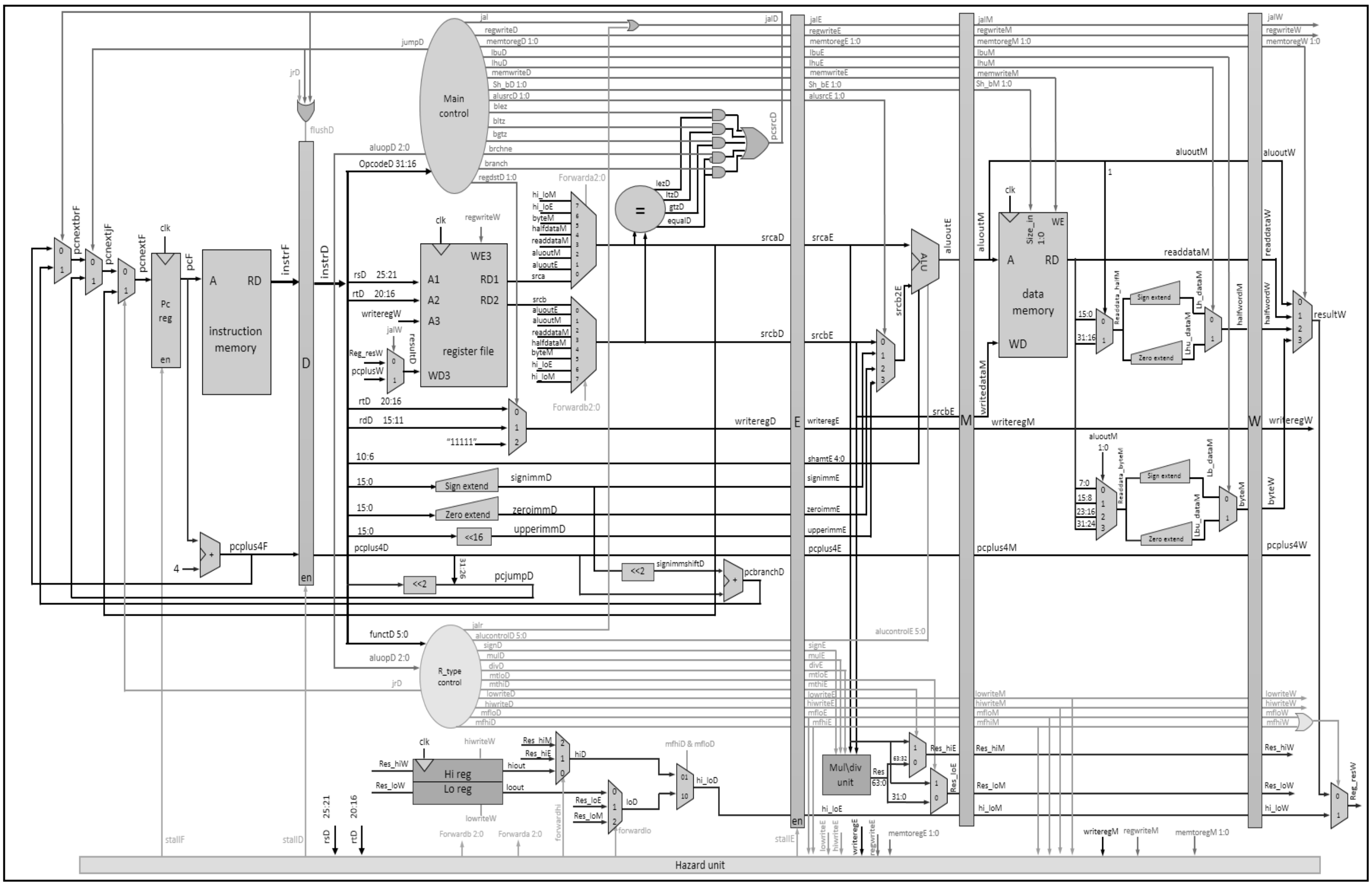


Figure 1: Complete Design of a Pipelined MIPS

After adding necessary electronic circuitry for each instruction execution, the final MIPS processor design supports all the 50 integer instructions.

Pipelined Datapath Unit

The biggest delays in the single-cycle processor are caused by reading and writing the memories and register file and using the ALU (**A**rithmetic / **L**ogic **U**nit) [1]. Therefore, to form the 5-stages pipelined processor, the single-cycle datapath must be chopped into five stages so that each stage performs exactly one of these slow steps.

Whenever a stage has finished its task on an instruction, it will pass it for the next stage and receive the following instruction. So pipeline registers are required to save the results of each stage. These registers are represented in Fig. 1 by long, dark green bars labeled as: **D, E, M,** and **W**.
These five stages are:

1. Fetch stage (**F**): in this stage, the processor uses the address of the instruction to execute contained in the PC (**P**rogram **C**ounter) register to read this instruction from instruction memory, and computes the address of the next instruction by incrementing the PC value by 4. This stage must contain the following components:
   - A 32-bit PC register.
   - A 32-bit word-addressable instruction memory.
   - An adder to increase PC by 4.
   - Multiplexers for selecting either branch address or jump address.
2. Decode stage (**D**): in this stage, the processor decodes the instruction to produce the control signals and reads the source operands from the register file. This stage must contain the following components:
   - A register file which consists of 32-bit * 32 registers.
   - A sign extender.
   - A zero extender.
   - Equal unit that makes the decision of branch instructions.
   - An adder and a 2-bit shifter to compute the target address in the case of jump and branch instructions.
   - Two 32- bit registers (*Lo* and *Hi*) to hold the results of **mul** and **div** instructions.
   - Multiplexers are used to select one input from several inputs and pass it to the output. Select line is controlled by a signal provided by the control unit.
3. Execute stage (**EXE**): in this stage, the processor uses either the ALU or the Mul/Div unit to perform the required computations which may be the calculation of values of two registers or address calculation for **lw** or **sw** instructions. EXE stage contains:
   - An ALU, which performs arithmetic and logic operations as described in table 1.
   - Mul/Div unit, which performs integer multiplication and division operations.
   - Multiplexers.
4. Memory stage (**MEM**): in this stage, the processor reads or writes data memory so only **lw** and **sw** instructions use this stage. MEM stage contains:

- A byte-addressable data memory.
- Sign and zero extenders.
- Multiplexers.

5. Write back stage (**WB**): in this stage, the processor writes the results back to the register file when needed. It only contains:
   - Multiplexors to choose which value to be written back to register file.

Table 1: Functions of ALU

Alucontrol (5:0)	function
000000	A and B
000001	A or B
000010	A + B
000011	Not used
000100	Sll
000101	A xor B
000110	A nor B
000111	Srl
001000	Sra
010000	A and B’
010001	A or B’
010010	A – B
010011	Slt
010100	Not used
010101	A xor B’
010110	A nor B’
100100	Sllv
100111	Srlv
101000	Srav

As noticed, the register file must be read in decode stage and written in write back stage within a single cycle, and this is possible only if the register file is written in the first part of a cycle and read in the second part as bellow:

```
if (clk'event and clk='0') then
   if (we3='1') then
       reg(CONV_INTEGER(a3)) <= wd3;
   end if;
end if;
```

Pipelined Control Unit

The processor recognizes the operation required by each instruction by examining the *opcode* and *funct* fields of the instruction in the decode stage to produce the control signals. These control signals must be pipelined along with the data so that they remain synchronized with the instruction. The pipelined control unit consists of:

1. Main control: this takes opcode (instrD 31:26) field as inputs and produces **multiplexer select**, **memory write** and **3-bit ALUop** signals as shown in table 2. The meanings of **ALUop** signals were given in table 3.

2. R-Type control: this uses **funct** (instrD 5:0) field of instruction with **ALUop** signals generated by the main decoder to produce **ALUcontrol (5:0)** signals and several signals that are necessary in the execution of R-type instructions. Table 4 shows the truth table of R-type control.

Table 2: Main Decoder Truth Table

Instruction	Opcode	Sh_B	Lbu	Lhu	Regwrite	Regdst	Alusrc	Branch	Brchne	Blez	Bltz	Bgtz	Memwrite	Memtoreg	Jump	Jal	Aluop
R_type	000000	xx	x	x	1	01	00	0	0	0	0	0	0	00	0	0	110
lw	100011	xx	x	x	1	00	01	0	0	0	0	0	0	01	0	0	000
sw	101011	11	0	0	0	xx	01	0	0	0	0	0	1	xx	0	0	000
beq	000100	xx	x	x	0	xx	00	1	0	0	0	0	0	xx	0	0	001
bne	000101	xx	x	x	0	xx	00	0	1	0	0	0	0	xx	0	0	001
blez	000111	xx	x	x	0	xx	00	0	0	1	0	0	0	xx	0	0	001
bltz	000001	xx	x	x	0	xx	00	0	0	0	1	0	0	xx	0	0	001
bgtz	000110	xx	x	x	0	xx	00	0	0	0	0	1	0	xx	0	0	001
addi	001000	xx	x	x	1	00	01	0	0	0	0	0	0	00	0	0	000
addiu	001001	xx	x	x	1	00	01	0	0	0	0	0	0	00	0	0	000
j	000010	xx	x	x	0	xx	xx	x	x	x	x	x	0	xx	1	0	xxx
jal	000011	xx	x	x	1	10	xx	x	x	x	x	x	0	xx	1	1	xxx
andi	001100	xx	x	x	1	00	10	0	0	0	0	0	0	00	0	0	010
ori	001101	xx	x	x	1	00	10	0	0	0	0	0	0	00	0	0	011
xori	001110	xx	x	x	1	00	10	0	0	0	0	0	0	00	0	0	100
slti	001010	xx	x	x	1	00	01	0	0	0	0	0	0	00	0	0	101
sltiu	001011	xx	x	x	1	00	01	0	0	0	0	0	0	00	0	0	101
lui	001111	xx	x	x	1	00	11	0	0	0	0	0	0	00	0	0	000
lb	100000	xx	0	0	1	00	01	0	0	0	0	0	0	11	0	0	000
lbu	100100	xx	1	0	1	00	01	0	0	0	0	0	0	11	0	0	000
lh	100001	xx	0	0	1	00	01	0	0	0	0	0	0	10	0	0	000
lhu	100101	xx	0	1	1	00	01	0	0	0	0	0	0	10	0	0	000
sb	101000	00	x	x	0	xx	01	0	0	0	0	0	1	xx	0	0	000
sh	101001	01	x	x	0	xx	01	0	0	0	0	0	1	xx	0	0	000

Table 3: ALUop Meaning

Aluop	Meaning
000	Add
001	Sub
010	And
011	Or
100	Xor
101	Slt
110	Look at funct field
111	N/a

Hazard Unit

Like any pipelined processor, this pipelined MIPS design suffers from three types of hazards:

1. **Structural hazards.**
2. **Data hazards.**
3. **Control hazards.**

Structural hazard happen when instructions compete for the same hardware resource. Memory is accessed by fetch (F) stage and memory (MEM) stage at the same clock cycle which led to a structural hazard. In this design, this hazard is solved by using two memories, one for data and the other for instructions.

During a program execution, some instructions will depend on the results if an instruction that did not have finished yet so this will lead to data hazards. Fig. 2 and Fig. 3 give brief ideas on how all data hazards are solved in this design:

- At clock cycle $\mathbf{t}_4$, if instruction $\mathbf{I}_4$ depends on results of instruction $\mathbf{I}_1$, no data hazard will happen. Since $\mathbf{I}_1$ has already reached the write back (**WB**) stage and written its result in the register file at the negative clock edge as shown in figure 2
- At clock cycle $\mathbf{t}_3$, if instruction $\mathbf{I}_3$ depends on results of instruction $\mathbf{I}_1$, forwarding will be possible. Since $\mathbf{I}_1$ has already reached the memory **(MEM)** stage and result has been calculated but has not been written back to register file yet as shown in figure 2.

Table 4: R-Type Decoder Truth Table

Aluop	Funct	Alucontrol	Jr	Jalr	div	Mult	Sign	Mthi	Mtlo	Mfhi	Mflo	Hiwrite	lowrite
000	xxxxxx	000010 (add)	0	0	0	0	0	0	0	0	0	0	0
001	xxxxxx	010010 (sub)	0	0	0	0	0	0	0	0	0	0	0
010	xxxxxx	000000 (and)	0	0	0	0	0	0	0	0	0	0	0
011	xxxxxx	000001 (or)	0	0	0	0	0	0	0	0	0	0	0
100	xxxxxx	000101 (xor)	0	0	0	0	0	0	0	0	0	0	0
101	xxxxxx	010011 (slt)	0	0	0	0	0	0	0	0	0	0	0
11x	100000	000010 (add)	0	0	0	0	0	0	0	0	0	0	0
11x	100001	000010 (add)	0	0	0	0	0	0	0	0	0	0	0
11x	100010	010010 (sub)	0	0	0	0	0	0	0	0	0	0	0
11x	100011	010010 (sub)	0	0	0	0	0	0	0	0	0	0	0
11x	100100	000000 (and)	0	0	0	0	0	0	0	0	0	0	0
11x	100101	000001 (or)	0	0	0	0	0	0	0	0	0	0	0
11x	100110	000101 (xor)	0	0	0	0	0	0	0	0	0	0	0
11x	100111	000110 (nor)	0	0	0	0	0	0	0	0	0	0	0
11x	101010	010011 (slt)	0	0	0	0	0	0	0	0	0	0	0
11x	101011	010011 (slt)	0	0	0	0	0	0	0	0	0	0	0
11x	000000	000100 (sll)	0	0	0	0	0	0	0	0	0	0	0
11x	000010	000111 (srl)	0	0	0	0	0	0	0	0	0	0	0
11x	000011	001000 (sra)	0	0	0	0	0	0	0	0	0	0	0
11x	000100	100100 (sllv)	0	0	0	0	0	0	0	0	0	0	0
11x	000110	100111 (srlv)	0	0	0	0	0	0	0	0	0	0	0
11x	000111	101000 (srav)	0	0	0	0	0	0	0	0	0	0	0
11x	001000	000010 (jr)	1	0	0	0	0	0	0	0	0	0	0
11x	001001	000010 (jalr)	1	1	0	0	0	0	0	0	0	0	0
11x	011000	000000 (mult)	0	0	0	1	1	0	0	0	0	1	1
11x	011001	000000 (multu)	0	0	0	1	0	0	0	0	0	1	1
11x	011010	000000 (div)	0	0	1	0	1	0	0	0	0	1	1
11x	011011	000000 (divu)	0	0	1	0	0	0	0	0	0	1	1
11x	010001	000000 (mthi)	0	0	0	0	0	1	0	0	0	1	0
11x	010011	000000 (mtlo)	0	0	0	0	0	0	1	0	0	0	1
11x	010000	000000 (mfhi)	0	0	0	0	0	0	0	1	0	0	0
11x	010010	000000 (mflo)	0	0	0	0	0	0	0	0	1	0	0

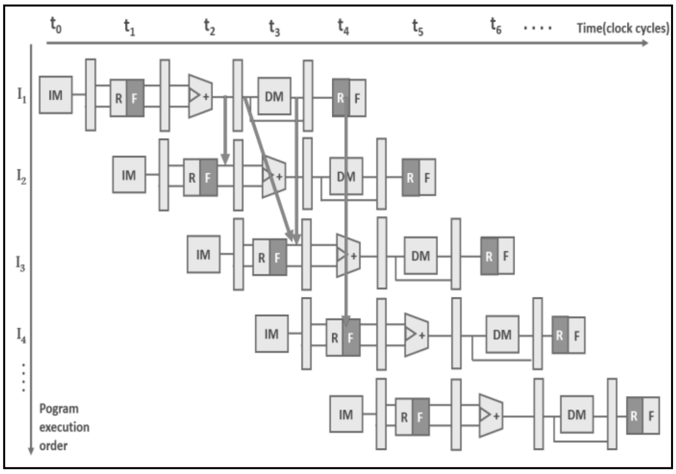


Figure 2: Forwarding to Solve Data Hazard

- At clock cycle t_2, if instruction I_2 depends on results of instruction I_1, two cases should be taken into consideration:

1. If instruction I_1 result comes from the execute (**EXE**) stage, the result has already calculated and forwarding is possible but the result has not been written back to register file yet as shown in figure 2.
2. instruction I_1 result comes from the memory (**MEM**) stage, the result has not been ready yet. The only solution is to stall the pipeline, holding up operation until the data is available as shown in figure 3.

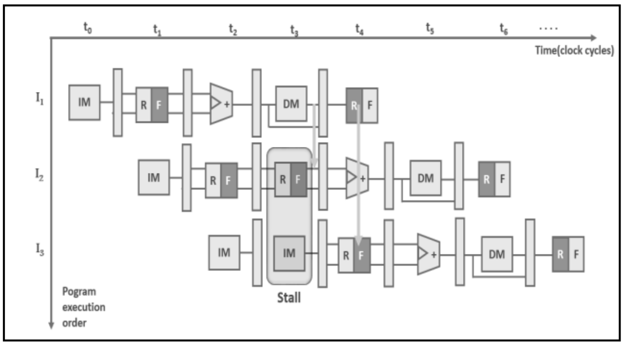


Figure 3: Stall to Solve Data Hazard.

The VHDL stall equations are:

stallF <= '0' when ((memtoregE = "01" or memtoregE = "10" or memtoregE = "11") and ((regwriteE ='1') and ((writeregE = rsD) or (writeregE = rtD)))) else '1';
stallD <= stallF;

The hazard unit shown in figure 4 along with four forwarding multiplexers are added to the pipelined processor so that it can solve data hazards with forwarding and stalls. The hazard unit generates control signals for the forwarding multiplexers as explained in tables 5 and 6.

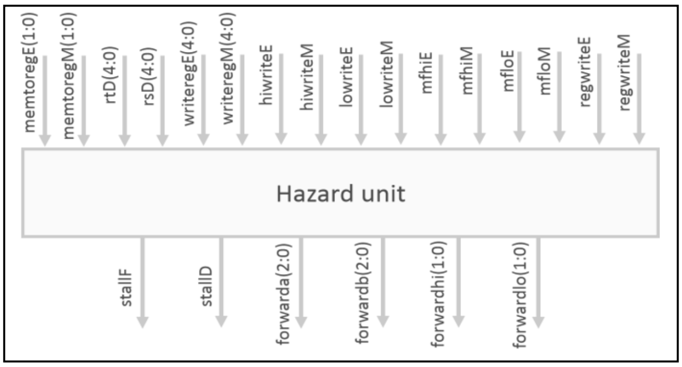


Figure 4: Hazard Unit

Control hazard happens when branches are needed. Depending on the branch decision that is made in the decode stage the processor does not know whether to fetch the following instruction or to take the branch. To reduce this hazard effect an always not taken static branch prediction is used in this design.
When a branch is executed, it is first fetched from instruction memory and then passed to the decode stage where the branch decision is made, at the same time, the following instruction is fetched. If decision is made and branch must be taken, the decode pipeline register is flushed and the instruction at the target address is executed otherwise continue executing the program in order.

Table 5: Hazard Unit Truth Table

Data source	Data destination	Condition	inputs: regwriteE	regwriteM	MemtoregM	mfhiE	mfloE	mfhiM	mfloM	outputs: forwarda	forwardb
Execute stage (EXE)	Decode stage (D)	(rsD ≠ "00000") and (rsD = writeregE)	1	x	xx	0	0	x	x	001	xxx
			1	x	xx	1	0	x	x	110	xxx
			1	x	xx	0	1	x	x	110	xxx
			1	x	xx	1	1	x	x	110	xxx
memory stage (MEM)	Decode stage (D)	(rsD ≠ "00000") and (rsD=writeregM)	x	1	00	x	x	0	0	010	xxx
			x	1	01	x	x	0	0	011	xxx
			x	1	10	x	x	0	0	100	xxx
			x	1	11	x	x	0	0	101	xxx
			x	1	00	x	x	1	0	111	xxx
			x	1	00	x	x	0	1	111	xxx
			x	1	00	x	x	1	1	111	xxx
Execute stage (EXE)	Decode stage (D)	(rsD ≠ "00000") and (rsD = writeregE)	1	x	xx	0	0	x	x	xxx	001
			1	x	xx	1	0	x	x	xxx	110
			1	x	xx	0	1	x	x	xxx	110
			1	x	xx	1	1	x	x	xxx	110
memory stage (MEM)	Decode stage (D)	(rsD ≠ "00000") and (rsD=writeregM)	x	1	00	x	x	0	0	xxx	010
			x	1	01	x	x	0	0	xxx	011
			x	1	10	x	x	0	0	xxx	100
			x	1	11	x	x	0	0	xxx	101
			x	1	00	x	x	1	0	xxx	111
			x	1	00	x	x	0	1	xxx	111
			x	1	00	x	x	1	1	xxx	111

The VHDL code for the flush signal is:

flushD <= pcsrcD or jumpD or jrD;

Table 6: Forward Signals for Mul/Div Unit

Data source	Data destination	hiwriteE	lowriteE	hiwriteM	lowriteM	forwardhi	forwardlo
Execute stage (EXE)	Decde stage (D)	1	x	x	x	01	xx
		x	1	x	x	xx	01
memory stage (MEM)	Decde stage (D)	0	x	1	x	10	xx
		x	0	x	1	xx	10

RESULTS

Figure 5 shows a program that finds the factorial of number 6. This program should write (0)h to memory location (84)h and (2d0)h to memory location (80)h if all instructions run correctly, if not, it means the VHDL design of the 5-stages, pipelined MIPS processor is incorrect.

After executing the test program by the 5-stage, pipelined MIPS processor, results that shown in figure 6 have been gotten. These results indicate the correctness of program execution which in turn reflects the correctness of the processor hardware design.

After that, a comparison between the single-cycle MIPS and the 5-stages, pipelined MIPS is made in terms of performance, speedup and FPGA utilization area as shown in table 7. The CPI (Clock Per Instruction) metric is calculated by using equation:

$$\textit{Program Execution time} = \textit{Instruction count} \times CPI \times \textit{Clock period}$$

Where instruction count in the test procedure is 90 instructions.

If successful, the value 0x2d0 is written to address 0x80 and 0x0 to address 0x84

	Assembly	Address	Description	Machine
Main:	addi $a0,$0,6	0	[$a0] = 0x6	20040006
	addi $sp,$0,0xfc	4	[$sp] = 0xfc	201d00fc
	jal factorial	8	should be taken, [$ra] = pc + 4	0c000006
	sw $v0,0x80($0)	c	[0x80] = [$v0] = 0x18	ac020080
	sw $v1,0x84($0)	10	[0x84] = [$v1] = 0x0	ac030084
Factorial:	addi $sp,$sp,-8	14	[$sp] = [$sp] - 0x8	23bdfff8
	sw $a0,4($sp)	18	[$sp + 4] = [$a]	afa40004
	sw $ra,0($sp)	1c	[$sp + 0] = [$ra]	afbf0000
	addi $t0,$0,2	20	[$t0] = 0x2	20080002
	slt $t0,$a0,$t0	24	[$a0] < [$t0] ? [$t0] = 1 : [$t0] = 0	0088402a
	beq $t0,$0,else	28	If [$t0] = 0 go to else	11000003
	addi $v0,$0,1	2c	[$v0] = 0x1	20020001
	addi $sp,$sp,8	30	[$sp] = [$sp] + 8	23bd0008
	jr $ra	34	should be taken, pc = [$ra]	03e00008
else:	addi $a0,$a0,-1	38	[$a0] = [$a0] - 1	2084ffff
	jal factorial	3c	should be taken, [$ra] = pc + 4	0c000006
	lw $ra,0($sp)	40	[$ra] = [$sp+0]	8fbf0000
	lw $a0,4($sp)	44	[$a0] = [$sp+4]	8fa40004
	addi $sp,$sp,8	48	[$sp] = [$sp] + 8	23bd0008
	mult $v0,$a0	4c	{ [hi] , [lo] } = [$v0] × [$a0]	00440018
	mfhi $v1	50	[$v1] = [hi]	00001810
	mflo $v0	54	[$v0] = [lo]	00001012
	jr $ra	58	should be taken, pc = [$ra]	03e00008

Figure 5: Test Procedure

Table 7: Comparison between Single-Cycle MIPS and Pipelined MIPS

processor	Program execution time	No. of clock cycles	Clock period	CPI	Speedup	FPGA area (Number of Slice Registers)
Single-cycle MIPS	1800 ns	90	20 ns	1	1	2,208
Pipelined MIPS	1125 ns	112.5	10 ns	1.25	1.6	2,946

This design is configured on *Xilinx Spartan-3AN* FPGA (**F**ield **P**rogrammable **G**ate **A**rray) starter kit and the results are shown in figure 7.

CONCLUSION

In this research, a 5-stages, 32-bit, pipelined MIPS processor has been designed in VHDL and synthesized using (Xilinx ISE design suite 13.4) program. This design consists of five pipeline stages which are: Fetch stage, Decode stage, Execute stage, Memory stage and Write back stage. It also solves all the problems of data hazard with forwarding and stalls and solves all the control hazard problems using always not taken static branch prediction with flushes.

After completing the design, several programs were executed and the desired results were obtained which indicate the processor ability to execute the designed 49 instructions.

Finally, it is recommended to adopt this research by universities to be used in a computer architecture course where students will be able to define, design, implement, and debug a complete processor, or a processor with specific instructions to be used for a dedicated purposes.

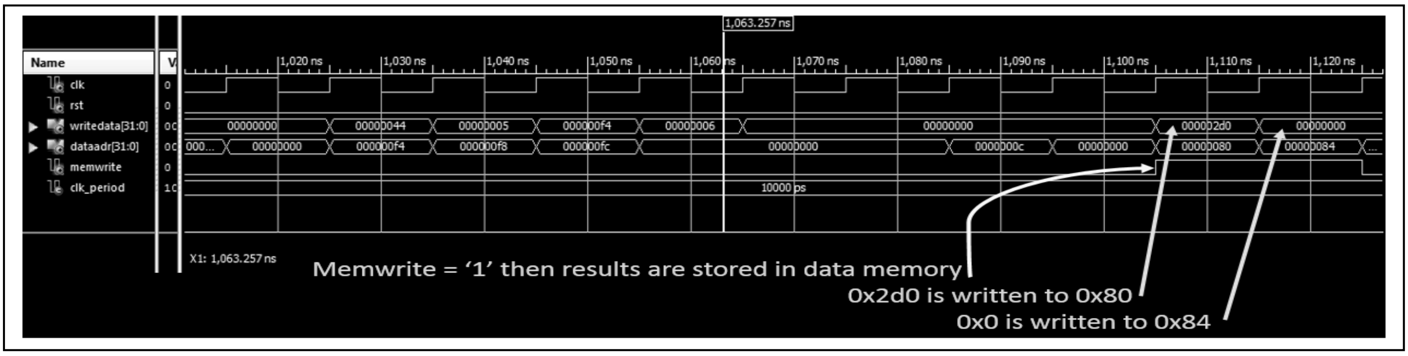


Figure 6: Test Program Execution

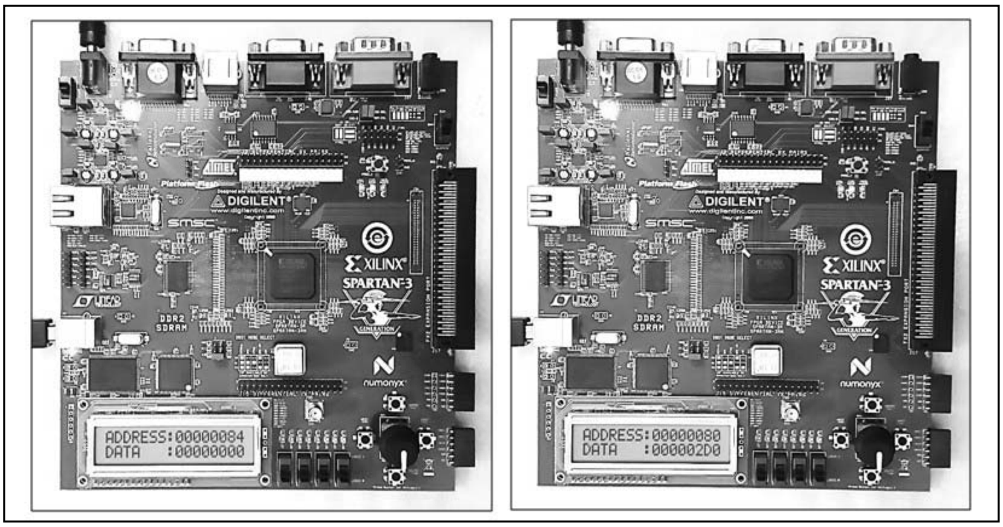


Figure 7: Results on SPARTAN-3AN

BIOGRAPHIES

SAFAA S OMRAN was born in Baghdad, Iraq. He received the B.Sc. Higher Diploma and M.Sc. in Electrical Engineering from Baghdad University, Iraq, in 1978,1981 and 1983 respectively. Since 1979 he joind the Foundation of Technical Education and worked at different Institutes of Technology. Now, he is an assistant prof. in the College of Elect. & Electronic Techniques, Baghdad, Iraq.

HADEEL MAHMOOD was born in Maysan, Iraq and earned her M.Sc. Degree in Computer Engineering Techniques from the College of Electrical and Electronic Techniques, Baghdad, Iraq, where she worked with asst. prof. Safaa Omran on VHDL Implementation of a RISC Processor. She is currently a lecturer assistant in the College of Electrical and Electronic Techniques, Baghdad, Iraq.

REFERENCES

Alekya, N. and P. G. Kumar. 2011. "Design Of 32-Bit RISC CPU Based On MIPS", *Journal of Global Research in Computer Science*, vol. 2, no. 9, 20-24, Sept.

Hennessy, J. L. and D. A. Patterson. 2012. *Computer Architecture: A Quantitative Approach*. 5th ed., San Francisco, USA: Morgan Kaufmann.

Hennessy, J. L. and D. A. Patterson. 2012. *Computer Organization and Design: The Hardware/Software Interface*, 4th ed., Waltham, USA: Morgan Kaufmann.

Linder, M. and M. Schmid. 2007. "Processor Implementation in VHDL", M. Eng. thesis, University of Ulster, Augsburg, Germany.

Omran, S. S. and H. Sh. Mahmood. 2013. "Hardware modelling of a 32- bit, single cycle RISC processor using VHDL." *ICIT 2013 The 6th International Conference on Information Technology,* Amman, Jordon, paper 632.

Pedroni,V. A. 2004. *Circuit Design with VHDL*, London, England: MIT Press.

Perry, L. 2002. *VHDL: Programming by Example*, 4th ed., America: McGraw-Hill.

Robio, V. 2004. "*A FPGA Implementation of A MIPS RISC Processor for Computer Architecture Education*." M. Eng. thesis, NewMexico State University, Las Cruses, New Mexico, America.

Singh, K. P. and S. Parmar. 2012. "Vhdl Implementation of a MIPS-32 Pipeline Processor." *International Journal of Applied Engineering Research*, vol. 7, No.11, 1952-195.

NETWORK SIMULATION

ADAPTIVE BANDWIDTH RESERVATION TECHNIQUE USING CROSS-LAYER APPROACH FOR MOBILE AD-HOC NETWORKS

Robin Rohit Vincent
Research Scholar
robinrohit@gmail.com

Ramraj Easwaran
Professor & Director
eramraj62@gmail.com

Alagappa University, Karaikudi, Tamilnadu, India

KEYWORDS
TDMA, CSMA, bandwidth estimation agent, cross layer design, MANET, channel assignment technique

ABSTRACT

In Mobile Ad hoc Networks (MANETs), inefficient resource allocation causes heavy losses to the service providers and results in inadequate user proficiency. For improving and automating the quality of service of MANETs, efficient resource allocation techniques are required. Cross platform is one other requirement in future generation of network. In this paper, we propose a Adaptive Bandwidth Reservation technique using Cross-layer Approach for Mobile Ad-hoc Networks. This approach adaptively estimates the bandwidth depending on the type of MAC protocol. It involves a Bandwidth Estimation Agent (BEA) that resides in each node and interacts with MAC and routing layers to perform the following activities. In resource reservation technique, the service provider is acting in two different ways for CSMA and TDMA modulation. Simulation results show that the proposed technique improves the throughput and received bandwidth.

INTRODUCTION

Mobile Ad hoc Network (MANET)

A Mobile Ad hoc Network (MANET) is a group of autonomous wireless nodes that may move randomly, forming a temporary network without any fixed backbone infrastructure. It has self-organized mobile nodes that are connected with relatively low bandwidth wireless links. Each node has its own area of influence that is called a cell, only within which others can receive its transmissions. In a MANET, there are no fixed infrastructures so nodes are free to roam. The network topology may change rapidly and unpredictably over time and nodes automatically made their own cooperative infrastructures [1] [2] [3]. In MANET network is decentralized, where all network activity including discovering the topology and delivering messages must be executed with the nodes it solves. There are various applications of MANET are video conferencing, rescue operations, military applications, Disaster Management etc. [2]

Bandwidth Reservation in MANET

Limited bandwidth provided by wireless channels in single-path routing is difficult to guarantee high traffic flows. Hence, multi-path routing improves end-to-end throughput over single-path routing by constructing multiple paths between a source-destination pair, and distributing high traffic load into multiple paths for parallel transmission. Therefore, congestion is alleviated and node’s energy consumption, the probability of a network partition and topology change is lowered. Hence, bandwidth assurance is necessary to transmit high traffic [4] as it causes heavy losses to the service providers and results in inadequate user proficiency. For improving and automating the QoS, efficient resource allocation techniques are required [5]. Bandwidth reservation guaranteeing a certain amount of bandwidth for a certain flow or service class requires that the station providing that guarantee is in control of that bandwidth [6].

Unlike in cellular networks where resources are controlled by a centralized entity, bandwidth reservation in ad-hoc networks is a more challenging task due to the instability of radio channels. Node mobility and lack of coordination between mobile nodes reservations established by a node should be reported to all its neighbors. Any missed or unrecorded reservation can have drastic impacts on the consistency of established reservations [7]. The provision of QoS slightly relies on resource/ bandwidth reservation [8].

Issues of Bandwidth Reservation in MANET

Bandwidth reservation issues are given below in detail:

- The active nature of the MANET causes unpredicted faults to disturb the bandwidth reservation.
- If traffic varies significantly, then the static bandwidth reservation is inadequate or under-exploited. [7]
- Occasionally some nodes cannot receive reservations packets sent by their neighbors due to collision of packets

at neighbor nodes or due to neighbor nodes that are transmitting control packets simultaneously [9].

- A node cannot hear a reservation packet of its neighbors when two neighbor nodes are transmitting at the same time. [9]
- For a transmission to take place, the receiver needs that no interference occurs during the whole transmission. Therefore, the value of the available bandwidth on a link depends on both peer channel utilization ratios and on the idle period synchronization. This issue is known as the channel's idle period synchronization issue.
- No collision detection is possible in MANET environment. Therefore, whenever a collision happens colliding frames are completely emitted maximizing the bandwidth loss. The collision probability is to be estimated and integrated to available bandwidth.
- When collisions happen on unicast frames, then ignoring back off account provides high inaccuracy in the estimated available bandwidth. So back off time is also an issue in bandwidth reservation.
- The node is first needed to contend for medium access whenever it sends frame in the network. If the channel is free, it can send a frame. Therefore, a sender needs to evaluate the proportion of time the medium is idle to determine the chance to gain access to the shared resource. So the carrier sense mechanism prevents two close emitters from transmitting simultaneously, unless they draw the same back off counter value. Therefore, an emitter shares the channel bandwidth with all its close neighbors. The expected delay should be computed for accurate bandwidth estimation. So this issue is known as Carrier sense mechanism issue [10]
- For some applications, such as real time and multimedia, need not only the capability to establish communications between nodes but also require of the network QoS guarantees on bandwidth, bit error rate, and delay. So bit error rate and delay are also an issue for bandwidth reservation.
- The calculations of bandwidth are complicated by the hidden and exposed terminal problems so there will be a problem with bandwidth reservation .[11]
- A transmission between two nodes does not consume bandwidth from the whole neighborhood; since no other neighbor will be able to transmit at the same time (at least using the same channel) in order to avoid collisions. So it is also an issue [12]

There is need for bandwidth estimation technique that considers both channel state and incoming and outgoing flow rate. It should to adaptable both TDMA and CSMA based MAC protocols providing a cross-layer interaction in the routing protocol. This paper designs an adaptive bandwidth reservation using cross-layer approach that adaptively estimates the bandwidth depending on the type of MAC protocol. The technique given in this paper involves a Bandwidth Estimation Agent (BEA) that resides in each node and interacts with MAC and routing layers to perform the following activities.

The paper work starts with a suitable introduction describing about issues and bandwidth reservation technique in the section one. In second section; this paper is giving literature work that supports this paper's proposed methodology. This paper describes a suitable proposed method with detail architecture and problem definition that is described in section three. The last section carries a suitable conclusion for this paperwork.

LITERATURE REVIEW

WENJING YANG et al., [4] have proposed a Bandwidth aware Multipath Routing (BMR) protocol, since the limited bandwidth of wireless networks has become the principal factor that restrains the development of multimedia applications. So given mechanism can find two disjoint parallel paths between source and destination based on the available bandwidth restriction and provides a metric for routing discovery. The performance metrics are end-to-end throughput, packet delivery ratio and end-to-end delay. The drawback of the paper is that using multi-path. The transmission increases throughput when compared with the single - path, but hidden terminal contention at destination lowers this throughput. So increasing the degree of end-to-end throughput decreases as the number of parallel paths grows

Rafael Guimarães et al [12] have described a QoS reservation mechanism for Multirate ad hoc wireless networks that allows bandwidth allocation on a per flow basis. In Multirate networks, wireless nodes are able to dynamically switch among several link rates. This allows nodes to select the highest possible transmission rate for exchanging data, independently for each neighbor. They provided a set of QoS constraints that must be satisfied for the ongoing QoS flows to consume an overall bandwidth at any node smaller than or equal to a certain threshold. They have applied the reservation scheme to the optimized link state routing protocol. Their solution not only guarantees QoS levels, but also distributes the traffic more evenly among network nodes.

Xu-Zhenet et al., [13] have proposed an efficient distributed timeslot assignment algorithm. This is an effective heuristic algorithm for calculating end-to-end bandwidth on a path, which can be used together with AODV to setup multi-path QoS routes. This algorithm increases network throughput, request success ratio and packet delivery ratio, and reduce end-to-end average packet delay. The drawbacks of the paper are that in given algorithm is overall time complex.

Maria Canales, et al., [15] have proposed adaptive cross-layer architecture based on the cooperation between QoS routing protocol and the MAC level. Their proposed scheme has been designed to perform flexible parameter configuration that allows adapting the system response to the observed grade of mobility in the environment. The performance metrics are Throughput, packet delivery. The drawback of the paper is that for cross-layer implementation, in the mobility situation, the variation in the admission conditions may lead to the result of throughput falls. It means reserved resources or the established links may be disconnected. It happens due to an increase in the packet discarding prompted by additional queuing during rediscoveries or the required retransmissions after collisions.

Wang Xiangli,. et al., [16] have proposed a distributed bandwidth reservation protocol (DBRP) for QoS routing in ad hoc networks. Their mechanism provides an efficient bandwidth reservation for QoS in ad hoc networks. It can effectively reserve bandwidth, control overhead and improve request success rate. The performance metrics are packet delivery and packet delay. The drawbacks of the paper are that given mechanism (DBRP) is not energy efficient and there is no discussion about throughput metric since throughput plays important role in MANET for QoS.

PROPOSED SOLUTION

Overview

Generally, the network layers like presentation layer, secession layer and are independent of handoff technologies. So, this proposal is only considering network layer, transport layer, MAC layer (data link layer) and physical layer. The author has given the requirements and description of the network according to the network layer.

Here, there are a number of service users and a MAC protocol present. The MAC protocol is getting the type of communication or modulation to be done every specific end node or mobile devices. There are two phases present for bandwidth estimation and channel allocation. Both the phases are working independently and parallel with different end node or mobile device.

Entries In Node's Routing Table- (BEA)

Each node constitutes the routing table that includes the entries of its id, flag, power level, node activating counter, cumulative assigned rates for incoming and outgoing flow and requested data rate stored in the bottleneck bandwidth (BWBN) field.

The amount of quantity of the routing table entries is found based on the number of the active incoming and outgoing flows that is expressed as n (n-1), where n is the number of neighbors of the node. The routing table also includes the following values

Assign ACA_{ij} corresponding to incoming and outgoing flow. Counter CNT_{ij} for the number of bits that have arrived in the current measurement window. Measure the rate of CA_{ij} from the previous measurement window. Every node is responsible for policing the incoming and outgoing flow to the cumulative assigned rate ACA_{ij} . This measured rate ACA_{ij} helps in performing rate-adjustment. The source node selects a path with minimum power consumption and congestion as per the previous paper (12)]. The following section describes the steps involved in the bandwidth reservation technique.

TDMA Virtual Slot Scheduling Phase

If the underlying MAC protocol is TDMA, it finds the available TDMA slots that are used to transmit in every link along the path. The BEA interaction with MAC layer to find these available slots and using a cross layer approach passes this information to the routing algorithm. The link bandwidth is then estimated from the TDMA slots using equation one and two. The TDMA slots are allocated using the TDMA Virtual Slot scheduling (VSLOT) approach [14].The below architectural diagram is showing that there are a number of service users are those working on TDMA. After detecting the modulation technique, the single service provider is estimating bandwidth and allocating slots to each end node or service provider.

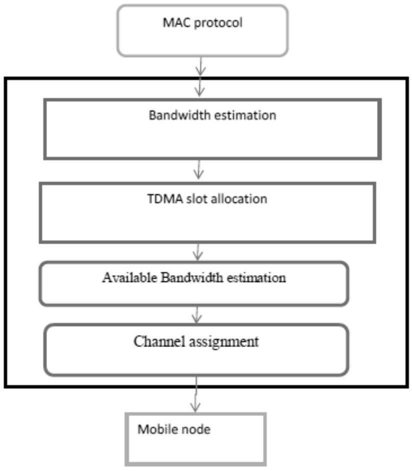


Figure 1 : The Architectural Diagram of TDMA Virtual Slot Scheduling Phase

Available Bandwidth Estimation

Every node is in charge for estimating the available bandwidth on its link. For a given node,

Let Bav = available bandwidth.

L = link capacity associated with one-hop neighbor i.

ACA is the cumulative assigned rates for all incoming and outgoing flows. Hence the sum of the assigned incoming and outgoing flow rates and available bandwidth on the link should be equal the capacity of the link i. This can be expressed as

$$ACA_{ij} + Bav_i = L_i \quad (1)$$

The mobile agent from the source node forwards the data packet along a given path towards the destination. The data packet constitutes the requested bandwidth value stored in the bottleneck bandwidth filed. Each intermediate node is responsible for determining whether or not sufficient bandwidth is available on the local outgoing link to support the new flow request. The link capacity is measured and available bandwidth is defined by Bav_j

$$Bav_j = \max\{0, L_j - ACA_{ij}\} \quad (2)$$

Algorithm for TDMA phase

Begin

Steps-1- register the mobile node or end node as service user of TDMA at a service provider.

Steps-2- the service provider is calculating the efficient bandwidth through the equation-1 and equation two.

Steps-3- bandwidth assignment is done through equation-3 and equation-4.

End

BEA And Available Bandwidth Estimation for 802.11 MAC Protocol

The below diagram is the architectural diagram of the second phase of the proposed method. In this phase, bandwidth estimation is done and it is applied to bandwidth estimation algorithm. This phase is for the IEEE 802.11 CSMA MAC protocol protocols. Here a single service provider and multiple service users are present. The service provider is going through a number of processes for bandwidth estimation and assigns channel to the end node or mobile device.

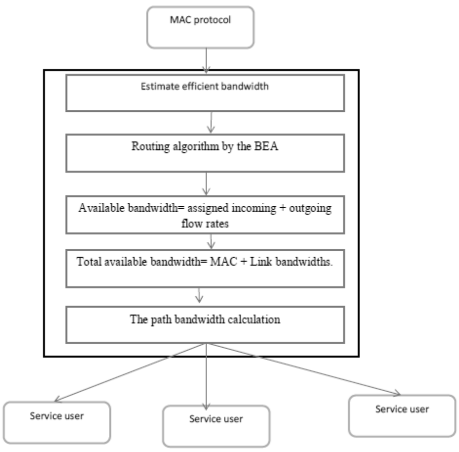


Figure 2 : The Architecture Diagram of BEA and Available Bandwidth Calculation for MAC protocol

If the underlying MAC protocol is IEEE 802.11, the available bandwidth can be estimated by using the channel status of the radio [7]. The BEA constantly monitor when the channel state changes; it starts counting when channel state changes from busy state to idle state and stops counting when channel state changes from idle state to busy state. The Idle Time (Ti) is composed of several idle periods during an observation interval t; the node adds all the idle periods to compute the total idle time. Then this information will be passed on to the routing algorithm by the BEA and the available bandwidth can be computed.

.

Estimation of link bandwidth

The total bandwidth on the link can be computed as the sum of the assigned incoming and outgoing flow rates and it should be equal to the capacity of the link.

The below formula is given for available bandwidth calculation.

$$BW_t=OF+IF \tag{3}$$

where BW_t= total bandwidth

OF= outgoing flow

IF= incoming flow

$$BW_t=L_{tc} \tag{4}$$

Where L_{tc}= link bandwidth.

Estimation Total Available Bandwidth

The total available bandwidth in a node is then given by adding the MAC and Link bandwidths.

$$BW_{avt}= L_{tc}+ BW_{MAC} \tag{5}$$

where BW_{avt}= total available bandwidth

BW_{MAC}= MAC bandwidth

The path bandwidth is then calculated towards the destination node by the routing protocol, and the calculated value represents the maximum available bandwidth between the source and the destination

The bandwidth is then allocated for the flows based on the estimated available path bandwidth.

Bandwidth Assignment

This paper assumes that the channel capacity in a network is C (channel capacity can be a reflection of loss exponent or power of a network). For simplicity this paper considers one way link here. This paper expresses bandwidth at any node i is

$$\sum\nolimits_N B_i \le \frac{C}{a} * N \tag{6}$$

where N is the number of nodes that shares the channel and 1/a is termed as channel utilization rate. It is quite obvious that as the number of nodes in an area increase (node density) channel will be shared by that much number of nodes or in other terms utilization par node will decrease. Hence this paper can devise a theory that a Node Density. This paper chooses a=Node Density by optimizing the proportionality constant to 1 as it will not have a direct effect on the performance. Based on this paper can say that bandwidth at any node I am

$$B_i = \frac{C}{a} * \frac{1}{n_i - u_i} \tag{7}$$

where n_i is the total neighbors of I and Ui is the used bandwidth by node i. This bandwidth measurement is very significant in jitter management as a node only needs to know about it's neighbors and can still estimate the available bandwidth. MAC layer will directly notify this bandwidth information to routing layer and the application layer. Hence the rate of packet transmission is adjusted by the top layer so that U_i is always maximum or bandwidth availability and usage is persistent. Since jitter and bandwidth are made independent, the network can predict the jitter.

Algorithm for bandwidth estimation phase

Begin
Steps-1- calculate the bandwidth as given in equation-1 and equation-2.
Step-2- get the total bandwidth from equation-3 and assign it as link bandwidth as given in equation-4.
Step-3- get the total bandwidth as given in equation-5.
Step-4- The path bandwidth is then calculated towards the destination node by the routing protocol and the calculated value represents the maximum available bandwidth between the source and the destination
Step-5- The bandwidth is then allocated for the flows based on the estimated available path bandwidth through equation-6 and equation-7.
End

SIMULATION RESULTS

Simulation Model and Parameters

The Network Simulator (NS2) [17], is used to simulate the proposed architecture. In the simulation, 50 mobile nodes move in a 1000 meter x 1000 meter region for 50 seconds of simulation time. All nodes have the same transmission range of 250 meters. The simulated traffic is Constant Bit Rate (CBR).
The simulation settings and parameters are summarized in table.

No. of Nodes	50
Area Size	1000 X 1000
Mac	IEEE 802.11
Transmission Range	250m
Simulation Time	10,20,30,40 and 50 sec
Traffic Source	CBR
Packet Size	512
Sources	4
Routing Protocol	AODV
Rate	500,1000,1500,2000 & 2500kb
Initial Rate	100Kb

Performance Metrics

The proposed Adaptive Bandwidth Reservation using Cross-Layer approach is compared with the BMR technique [4]. The performance is evaluated mainly, according to the following metrics.

- **Received Bandwidth:** It is the number of bits received by the receiver.
- **Packet Lost**: It refers the average number of packets dropped during the transmission
- **Delay**: It is the amount of time taken by the nodes to transmit the data packets.
- **Packets Received**: It is the number of packets received by the receiver.

RESULTS

Case-1: For on IEEE 802.11 MAC protocol

We vary the rate as 500,1000,1500,2000 and 2500Kb

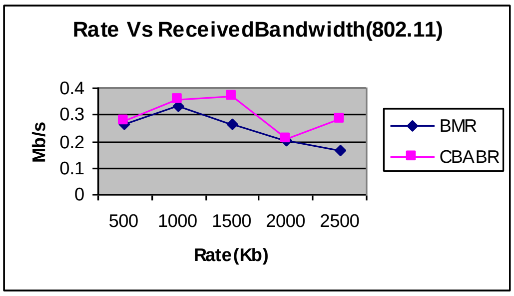


Figure 3: Rate Vs Received Bandwidth

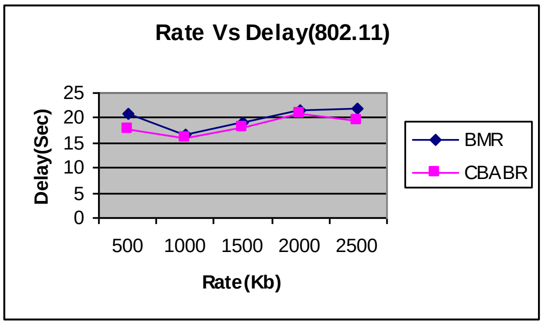


Figure 4: Rate Vs Delay

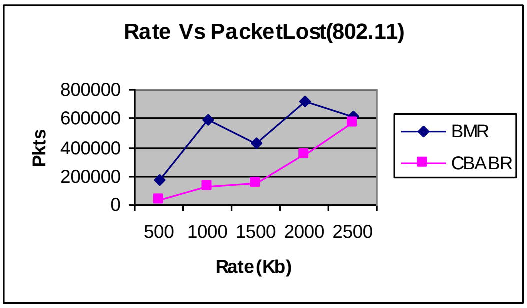


Figure 5: Rate Vs Packet Lost

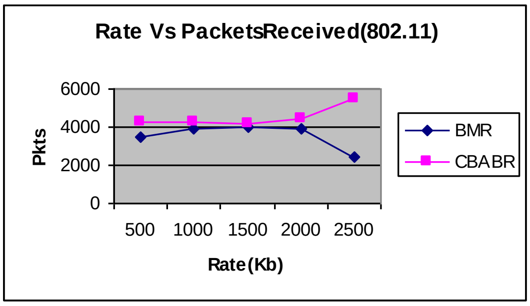


Figure 6: Rate Vs Packets Received

Figure 3 shows the received bandwidth of CBABR and BMR techniques for different rate scenario. We can conclude that the received bandwidth of our proposed CBABR approach has 17% of higher than BMR approach.

Figure 4 shows the delay of CBABR and BMR techniques for different rate scenario. We can conclude that the delay of our proposed CBABR approach has 7% of less than BMR approach.

Figure 5 shows the packets lost of CBABR and BMR techniques for different rate scenario. We can conclude that the packets lost of our proposed CBABR approach have 57% of less than BMR approach.

Figure 6 shows the packets received of CBABR and BMR techniques for different rate scenario. We can conclude that the packets received of our proposed CBABR approach has 19% of higher than BMR approach.

Case-2: For on TDMA MAC protocol

We vary the rate as 500,1000,1500,2000 and 2500Kb

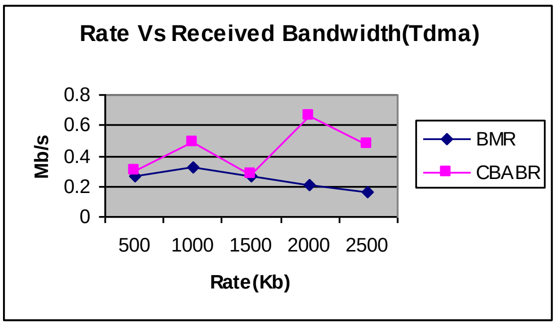


Figure 7: Rate Vs Received Bandwidth

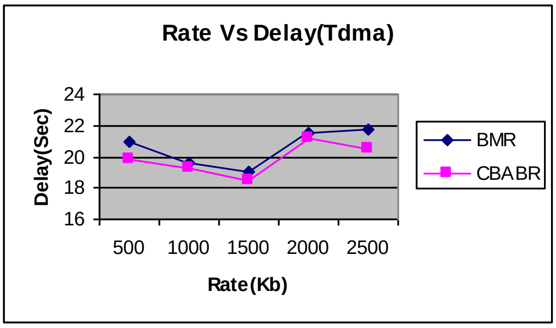


Figure 8: Rate Vs Delay

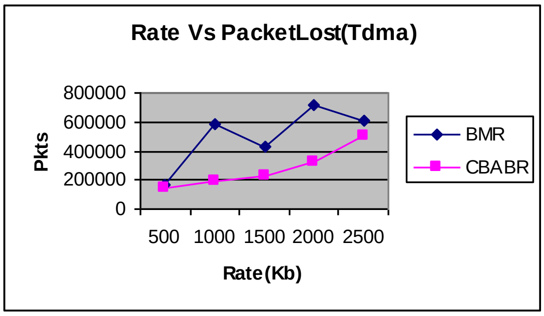


Figure 9: Rate Vs Packet Lost

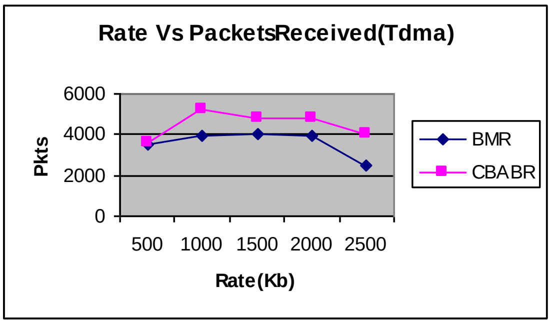


Fig 10: Rate Vs Packets Received

Figure 7 shows the received bandwidth of CBABR and BMR techniques for different rate scenario. We can conclude that the received bandwidth of our proposed CBABR approach has 37% of higher than BMR approach.

Figure 8 shows the delay of CBABR and BMR techniques for different rate scenario. We can conclude that the delay of our proposed CBABR approach has 4% of less than BMR approach.

Figure 9 shows the packets lost of CBABR and BMR techniques for different rate scenario. We can conclude that the packets lost of our proposed CBABR approach have 42% of less than BMR approach.

Figure 10 shows the packets received of CBABR and BMR techniques for different rate scenario. We can conclude that the packets received of our proposed CBABR approach has 20% of higher than BMR approach.

CONCLUSION

In this paper, an adaptive bandwidth reservation technique using cross-layer approach has been proposed that adaptively estimates the bandwidth depending on the type of MAC protocol. Two different methodologies have been devolved for two CSMA and TDMA techniques. This is a parallel done procedure. The computing is done at base stations. The estimation is done on user’s choice. Both the phases include channel bandwidth estimation and channel allocation techniques. The ultimatum goal of cross layer service is being described. To describe a suitable literature work is followed by an introduction. A brief description of all the proposed methodology has been done.

The proposed methodology can work on the multi-platform oriented system. The cross layer adoption is making the system more efficient for next generation network. The system is working on CSMA and TDMA techniques. Simulation results show that the proposed technique improves the throughput and received bandwidth when compared with existing techniques.

REFERENCES

Zhao Wen Xing and Le Gruenwald. 2013. “Issues in Designing Concurrency Control Techniques for Mobile Ad-hoc Network Databases”, *Distributed and Parallel Databases*, Vol. 31, Issue 2, DOI: 10.1007/s10619-012-7114-2.

Mukilan P and Dr. Wahi A. 2012. "EENMDRA: Efficient Energy and Node Mobility based Data Replication Algorithm for MANET", *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 3, No 1, May 2012. ISSN (Online): 1694-0814.

Aekyung Moon, Han Namgoong, and Haengrae Cho. 2004. "Energy-Efficient Replication Extended Database State Machine in Mobile Ad-Hoc Network, *Proceedings of IADIS International Conference on Applied Computing*, 224-228

Wenjing YANG, Xinyu YANG, et al. 2011. "A Bandwidth Aware Multi-path Routing Protocol in Mobile Ad Hoc Networks", *Journal of Computational Information Systems*, Binary Information Press, 685-696

Vishnu Kumar Sharma, Dr. Sarita Singh Bhadauria. 2011. "Agent based Bandwidth Reservation Routing Technique in Mobile Ad Hoc Networks". *International Journal of Advanced Computer Science and Applications*, Vol. 2, No. 12.

Javed Parvez and Mushtaq Ahmad Peer. 2010. "A Comparative Analysis of Performance and QoS Issues in MANETs", *World Academy of Science, Engineering and Technology*. ISSN 2250-2459, Volume 2, Issue 4, April 2012.

Sarita Singh Bhadauriaand Vishnu Kumar Sharma. 2012. "Bandwidth Reservation Routing Technique Based on Agent in Mobile Ad Hoc Networks Using Rate Control with AODV" IPCSIT vol. 30 (2012), IACSIT Press, Singapore

Sourabh Singh Verma, Saroj Kr. Lanka, et al. 2012. "Precedence Based Preemption and Bandwidth Reservation Scheme in MANET", *International Journal of Computer Science Issues*, Vol. 9, Issue 6, No 2.

Ghalem Boudour, Cédric Teyssié, Zoubir Mammeri. 2009. "A Robust Reservation Protocol for Wireless Ad-hoc Networks" IWCMC'09, June 21–24, 2009, Leipzig, Germany.

Mohideen Vahitha Banu K. 2010. "Discussion on Improving Quality of Service through Available Bandwidth Estimation in Mobile Ad Hoc Networks, *International Journal of Computer Application* (0975 – 8887) ,Volume 11– No.8, December 2010.

Imad Jawhar and Jie Wu. 2004. "A Race-Free Bandwidth Reservation Protocol for QoS Routing in Mobile Ad Hoc Networks", HICSS '04 Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 9 - Volume 9, Page 90293., IEEE Computer Society Washington, DC, USA, ISBN:0-7695-2056-1

Rafael Guimarães, Llorenç Cerdà, José M. Barceló, "Quality of service through bandwidth reservation on multirate ad hoc wireless networks", *Ad Hoc Networks, Volume 7, ELSEVIER*, Issue 2, March 2009, Pages 388-400, ISSN 1570-8705

Xu-Zhen, Huang-Chuan He, Hu-Xian Zhi. 2008. "Interference-aware multi-path routing and bandwidth allocation for mobile wireless networks," Communication Systems, ICCS 2008,11th IEEE Singapore International pp.1685,1689, 19-21 Nov. 2008. doi: 10.1109/ICCS.2008.4737468

Rekha Patil. R, Damodaram. A, Rupam Das R. 2009. "A cross layer approach for controlling jitter in MANET based on bandwidth estimation", Internet, AH-ICI 2009. First Asian Himalayas International Conference on , pp.1-5, 3-5 Nov. 2009, doi: 10.1109/AHICI.2009.5340274

Marıa Canale, Jose Ramon Gallego, et al. 2009. "QoS provision in mobile ad hoc networks with an adaptive cross-layer architecture", "Springer Science Business Media, Wireless Networks", November 2009, Volume 15, Issue 8, pp 1165-1187.

Wang Xiangli, Li Layuan, Gong Bencan. 2007. "A Distributed Bandwidth Reservation Protocol for QoS Routing in Mobile Ad Hoc Networks," *Wireless Communications, Networking and Mobile Computing*, 2007. WiCom 2007. International Conference on , pp.1689,1692, 21-25 Sept. 2007,doi:10.1109/WICOM.2007.425.

Rabia Ali and Dr. Fareeha Zafar. 2012. "Bandwidth Estimation in Mobile Ad-hoc Network (MANET)", *IJCSI International Journal of Computer Science Issues*, Vol. 8, Issue 5, No 1, September 2011 ISSN (Online): 1694-0814

Network Simulator : http://www.isi.edu/nsnam/ns

BIOGRAPHIES

ROBIN ROHIT VINCENT obtained his Bachelor's degree in Electrical and Electronics Engineering from M.S University, India. Then he obtained his Master's degree in Computer Science and Engineering. He has also obtained CCNA professional qualifications. Currently, he is a **Research Scholar** pursuing his PhD in Computer Science and Engineering majoring in Networking in Alagappa University, India.

DR. RAMRAJ EASWARAN is presently working as a Professor in the Department of Computer Science and Engineering at Alagappa University, Karaikudi, Tamilnadu, India. He has over 24 years of teaching experience and 5 years of research experience. He has presented more than 35 research papers in National and International Conferences and published more than 45 papers in National and International Journals. His research areas include Data mining and Network.

Yawsnp: Yet Another Wireless Sensor Protocol

Mokhtar Beldjehem
University of Ottawa, School of Electrical Engineering and Computer Science, 800 King Edward, Ottawa, Ontario, K1N 6N5, Canada
E-mail: mbeldjeh@uOttawa.ca

KEYWORDS
Wireless Sensor Network, Hierarchical Soft Clustering, Self-Organizing, Fault-Tolerant, Energy-Aware, Load Balancing, Ambient Intelligence.

ABSTRACT

This paper describes an energy-aware multi-hop cluster-based fault-tolerant load balancing hierarchical routing protocol for a self-organizing wireless sensor network (WSN), which takes into account the broadcast nature of radio. The main idea is using hierarchical fuzzy soft clustering enabling non exclusive overlapping clusters, thus allowing partial multiple membership of a node to more than one cluster, whereby for each cluster the clusterhead (CH) takes in charge intra-cluster issues of aggregating the information from nodes members, and then collaborate and coordinate with its related overlapping area heads (OAHs), which are elected heuristically to ensure inter-clusters communication. This communication is implemented using an extended version of time-division multiple access **(**TDMA) allowing the allocation of several slots for a given node, and alternating the role of the clusterhead and its associated overlapping area heads. Each cluster head relay information to overlapping area heads which in turns each relay it to other associated cluster heads in related clusters, thus the information propagates gradually until it reaches the sink in a multi-hop fashion.

I. INTRODUCTION AND MOTIVATIONS

Wireless sensor networks (WSNs) have emerged as an important application of the ad-hoc networks paradigm, and were extensively applied for monitoring physical environment, infrastructure security, structural health monitoring and industrial sensing.. In recent years, we have been witnessing the proliferation of WSNs [1-9]. It is easy to predict that in the near future, WSNs will play important roles in our society. Consequently, demands for WSNn with various protocols, architectures and abilities to flexibly run various tasks can be expected to increase significantly. We believe that WSNs is the key enabling technology that will provide the adequate infrastructure to what is called *Ambient Intelligence* (AmI)[10,11].
In WSNs, sensors can be deployed either *randomly* or *deterministically*. A random sensor placement may be suitable for battlefields or hazardous areas while a deterministic sensor placement is feasible in friendly and accessible environments. In general, fewer sensors are required to perform the same task with a deterministic placement. WSNs are typically deployed in hazardous or inaccessible environments and hence the sensor nodes' energy supply is usually limited and cannot be renewed. Due to these limitations, the nodes' energy consumption must be minimized, while still maintaining the network's connectivity to maximize its useful lifetime. The nodes communicate *wirelessly* and often *self-organize* after being deployed in an ad hoc fashion. These self-organizing sensor networks have limitations of system resources like battery power, communication range, memory space and processing capability. Low processing power and wireless connectivity make designing such networks a real challenge. *Self-organization* can be defined as the process by which systems tend to reach a particular objective with minimal human interference. The mechanisms dictating its behavior are internal to the system. Network self-organization: Given the large number of nodes and their potential placement in hostile locations, it is essential that the network be able to self-organize; manual configuration is not feasible. Moreover, nodes may fail (either from lack of energy or from physical destruction), and new nodes may join the network. Therefore, the network must be able to periodically reconfigure itself so that it can continue to function. Individual nodes may become disconnected from the rest of the network, but a high degree of connectivity must be maintained**.** *Scalability* requires that any configuration process be completely distributed and use only local information, which presents the classic problem confronting all self-organized systems: How to obtain global optimality from local adaptation.
The main task of a sensor node in a sensor network is to monitor events, i.e., collect data, perform quick local data *aggregation*, and then transmit the data. Power consumption can hence be divided into three domains: sensing, aggregation, and communication, of which communication has the lion's share. This paper proposes a new framework to conserve energy of WSN, thereby the lifetime of the network is increased.
The wireless sensor network consists of different sensor node that have very limited battery power, so the main objective is to maximize the lifetime of sensor network. In WSN the energy is basically consumed by data transmission, as approximately 70% of the energy is consumed by data transmission so data transmission should be optimized in wireless sensor network, for maximizing the lifetime of network. Data transmission can be optimized with the help of effective protocol and effective ways of data

fusion (*aggregation*). There is a need to devise protocols, which can reduce the transmission load and enhance the life time of the entire network. Low energy use: Since in many applications the sensor nodes will be placed in a remote area, service of a node may not be possible. In this case, the lifetime of a node may be determined by the battery life, thereby requiring the minimization of energy expenditure.

Many of the design challenges for large scale wireless sensor networks (WSN) are well known, among them: *scalability*, *geographic range*, *infrastructure*, *cost*, *longevity*, *heterogeneity*, and *mobility*.

Due to the limited processing power, and finite power available to each sensor node, regular ad-hoc routing techniques cannot be directly applied to sensor networks domain. Thus, energy-efficient routing algorithms suitable to the inherent characteristics of these types of networks are needed.

LEACH[12] is a cluster-based energy-aware routing protocol which is one-hop protocol where a cluster head is assumed to relay the information directly to the sink in a single hop; however due to range limitations and the higher power node-to-sink direct broadcast; multiple hops through network may be required. As it is the case in common applications.

To this goal, fuzzy and soft clustering could provide a novel multi-hops hierarchical cluster based routing energy-aware protocol that prolong the sensor network lifetime while ensuring robustness and fault-tolerance. The remainder of the paper is organized as follows: section II surveys related literature on WSN protocols; section III describes design issues of the proposed novel protocol and analyses its quality aspects; section V attempts to draw some conclusions and to suggest some research directions.

II. BACKGROUND AND RELATED WORK

II.1 WSNs Protocols

A WSN consists of a number of independent nodes that communicate with each other wirelessly over limited frequency and bandwidth. WSN nodes are densely deployed and coordinate with each other to produce high-quality information about the sensing environment. The exact location of a particular sensor is unknown. It means that sensor network protocols and algorithms must provide self-organizing capabilities. The topology of the WSNs can vary from a simple star network to an advanced multi-hop wireless mesh network. The propagation technique between the hops of the network can be routing or flooding. Routing protocols in sensor networks from network structure point of view can be divided into two main categories: *flat* and *hierarchical*. In flat routing protocols the concept of *leader node* (or a cluster head) does not exists and all nodes are at the same level of importance. In hierarchical routing protocols the act of clustering and classification of nodes are done and some nodes are considered as *leaders*(or a cluster heads). From this group of protocols we can name LEACH. Indeed there are other categories of protocols like *data centric*, *location based*, *energy awa*re. In a way that each routing protocol can belong to one or several of mentioned groups.

Networks can be broadly classified into two types depending on the way they transmit data, namely, *point-to-point* networks or *broadcast* networks. In the case of point to-point networks, a separate channel exists between two separate nodes. In contrast, in the case of a broadcast network, there is only one channel available which is shared by all nodes on the network. *Media access control* **(**MAC) protocols control access to this shared channel.

II.2 Flooding, Gossiping and Clustering

In *flooding*, each sensor node receiving a data or a control packet repeats it by broadcasting, flooding has several shortcomings such as *implosion*(duplicated messages are sent to the same node), *overlap*(neighbor nodes receive duplicated messages) and *resource blindness*(does not take into account the energy resources, it is not energy aware). In gossiping, nodes do not broadcast but send the incoming messages to a randomly selected neighbors. A sensor node randomly selects one of its neighbors to send the data. Once the neighbor node receive the data, it randomly selects another sensor node. Gossiping avoids the implosion problem but still is not energy aware, and it takes a long time to propagate the message to reach the sink.

One way to support efficient communication between sensors is to organize the network into several groups, called clusters, with each cluster electing one node as the cluster head (CH). Conventionally, nodes are often grouped into disjoint and mostly non-overlapping clusters. Many literatures are concentrated on finding solution at various levels of the communication protocol, including being extremely energy efficient. Energy efficiency is often gained by accepting a reduction in network performance. Low-energy adaptive clustering hierarchy (LEACH) is a new communication protocol that tries to distribute the energy load evenly among the network nodes by randomly rotating the cluster head among the sensors.

II.3 Overview and Description of LEACH

Efficient routing in a wireless sensor network requires that the routing protocol must minimize energy dissipation (maximize energy conservation) and maximize network life time.

LEACH is a cluster-based energy-aware routing protocol which is *one-hop* protocol where a *cluster head* (which contain a longer range radio) is assumed to relay the information directly to the sink in a single hop; however due to range limitations and the higher power node-to-sink direct broadcast; *multiple hops* through network may be required in some practical situations. The cluster head may be selected in a randomized manner. Such a randomized selection of the cluster head, combined with rotating the cluster head position, can effectively avoid the early drain of the energy of a particular node.

LEACH is a cluster-based protocol that uses time-division multiple access (TDMA) for intra-cluster communication between the sensors and the cluster-head. When clusters are formed, the cluster-head node creates a schedule that gives each sensor in the cluster a time slot in which to transmit its data, allowing the sensors to sleep for all other time slots. LEACH includes mechanisms to allow the high energy position of cluster-head to be rotated among the nodes to evenly distribute the energy load. Clusters are adapted based on the cluster-head nodes for each round, and new TDMA schedules are created based on the new clusters. LEACH has been shown to achieve good energy-efficiency and hence long network lifetime when sensors always have data to send and when sensors are static. However, if sensors enter or leave the cluster area or change their data rate due to detection of phenomena while the cluster is fixed, LEACH cannot adapt.

In common applications, a typical network configuration consists of sensors working unattended and transmitting their observation values to some processing or control center, the so-called sink node, which serves as a user interface. Due to the limited transmission range, sensors that are far away from the sink deliver their data through *multihop* communications, i.e., using intermediate nodes as relays.

This paper deals about the framework for energy conservation of a Wireless sensor network. The framework is developed such a way that the nodes are allowed to be clustered in overlapping clusters, electing the cluster head, electing the overlapping head, performing intra-cluster transmission, inter-cluster transmission and from the cluster head, through the overlapping head, to the neighbor cluster head the information is transmitted or rather propagated progressively until it reaches the base station (or sink) in a multi-hop way.

III. A NOVEL MULTI-HOP FAULT-TOLERANT AND LOAD BALANCING HIERARCHICAL PROTOCOL

III. 1 Motivations and Design challenges

Different routing protocols have been developed to deal with this problem. It is still a great challenge of the hierarchical routing protocol to operate efficiently in the presence of node failure. Therefore, a novel hierarchical routing protocol that addresses network survivability and redundancy issues is needed. Fault-resilient protocols require the possibility to reach the sink using multiple paths. The goal is to devise a high capacity WSN that degrades softly under attack, while always providing a critical level of service. We have the following goals for designing Multi-hop, Fault-tolerant and Load Balancing Hierarchical Protocol. (i) *Localized algorithms.* It should only use localized algorithms and not depend on global state information for data delivery. (ii) *Energy efficiency*. The data delivery scheme should be energy efficient as sensor nodes are resource constrained. (iii) *Scalability*. The scheme should be scalable to large and dense sensor networks that consist of thousands of nodes. *iv)Fault-tolerance* to node failure or death.

III. 2 Benefits of multi-hop hierarchical clustering

The clustering approach has proved to be one of the most effective mechanisms to improve energy efficiency in wireless sensor networks (see, e.g., [2–7]). In a cluster-based sensor network, sensor nodes are organized into groups, each with a cluster head(CH). Traditionally, sensor nodes in a cluster send their data to the corresponding cluster head, and the cluster head forwards the data to the neighboring cluster along the route or to the sink directly. Building on the cluster based model, we propose an overlapping-cluster network structure, where overlapping areas heads (OAH) sensor nodes can carry out inter-cluster cooperative data transmissions. This structure is motivated by the two key features of wireless sensor networks: *node cooperation* and *data correlation*, which differentiate wireless sensor networks from conventional wireless networks. The way of operating of the algorithm is similar but different from the conventional hierarchical cluster-based schemes.

There are two types of wireless cluster-based protocol WSN: *Single hop* in which nodes belonging to a given cluster transmit to the cluster head and in its turn can transmit or communicate directly with the Sink. All the nodes use the same channel to communicate, and the message broadcast by one of the stations on the common channel is simultaneously heard by all other stations. In the *multi-hop* wireless networks intermediate nodes are used to route message from the source to the Sink. There is a strong need for development of routing techniques which work considerably across wide range of applications. In this paper only multi-hop wireless networks are considered.

During the steady phase data packet sent by a sensor (sender) reaches all its cluster member nodes within the transmission range of the sender; sensors far from the data sink have to use intermediate nodes to relay data transmission to reach the sink.

III.3 Protocol Design Description

The proposed solution is appropriate for random deployment and suitable for different sizes of target areas. This protocol forms clusters in which each cluster member is at one hop distance from the cluster head. This protocol ensures the participation of all the cluster heads in hierarchical topology formation. The proposed protocol is also capable of handling dynamic nature of the wireless sensor networks. It is a *multi-path* routing scheme which is more robust than a single-path scheme. Three kinds of sensor nodes are involved, cluster member(CM), clusterhead(CH) and overlapping area head (OAH).

In particular, the cluster head (CH) carries out data aggregation and coordinates the overlapping areas heads(OAHs) sensor nodes but not necessarily transmits the data itself to the Sink, whereas in a traditional cluster the cluster head performs the bulk of the communication tasks.

III.3.1 *Design Tradeoffs*

We are interested at devising a multi-hop, multi-path fault-resilient and load balancing hierarchical protocol. A CH need to relay his data to the sink through a multi-hop chain

A CH has the possibility to relay his data through OAHs and other CHs to the sink through multi-path. The goal is to develop a cross layer architecture that provides energy preservation, resilience and scalability for WSNs. The resulting architecture will be able to adaptively provide the appropriate trade-offs between *performance*, and *fault-resilience(or fault-tolerance).* Resilience could be achieved through a multi-path topology of the WSN (*Self-healing*). Energy-preservation through cluster-based hierarchical routing protocol. Scalability through self-organizing distributed routing protocol.

III.3.2 *Hierarchical Fuzzy Soft Clustering*

We have devised an energy-aware multi-hop cluster-based fault-tolerant load balancing hierarchical routing protocol for a self-organizing wireless sensor network (WSN), which takes into account the broadcast nature of radio. As illustrated in Fig. 1, the main idea is using fuzzy soft clustering enabling non exclusive overlapping clusters, thus allowing partial multiple membership of a node to more than one cluster and nested clusters, whereby for each cluster the cluster head (CH) aggregate the information from nodes members, and then collaborate and coordinate with its overlapping area heads (OAHs) to ensure inter-cluster communication. This communication is implemented using an extended version of TDMA allowing the allocation of several slots for a given node, and alternating the role of the cluster head and its associated overlapping area heads. This protocol forms clusters in which each cluster member is at one hop distance from the cluster head. Each cluster head relay information to all of its overlapping areas heads which in turns each relay it to its associated cluster head, the information propagate gradually until it reaches the sink. The algorithm distributes the energy load evenly among the network nodes by randomly rotating the cluster head among the sensors at the cluster level as well the overlapping area heads at the overlapping zones levels. The proposed protocol is also capable of handling dynamic nature of the wireless sensor networks such as links change, nodes enter (new nodes) and leave (dead nodes). The algorithm is composed of a *set up phase* followed by *steady phase*: The set up phase is made up of three main steps; the first stage is to partition the network into k overlapping clusters and determine a CH for each cluster, the second stage is to find out OAHs in-between clusters and determine an OAH for each pair of two neighboring clusters, the third stage is to partition the CHs nodes around related OAHs, whereby an OAH play the role of a CH in respect to the CHs with which it is related

III.3.2 .1 *The set up phase:*

Identification of cluster heads (CHs)

Is similar to LEACH, a sensor node chooses a random number in the range [0, 1]. If this random number is less than the threshold T(n), the sensor node is a clusterhead (CH). T(n) is computed as:

$$T(n) = \begin{cases} \dfrac{P}{1 - P*(r \bmod (1/P))} & if \quad n \in G \\ 0 & otherwise \end{cases}$$

Where P is the desired percentage to become a clusterhead, r is the current round, and G is the set of nodes that have not being elected as a clusterhead in the last 1/P rounds.

Basically, first a cluster head ensures intra-cluster aggregation and communication within its cluster among the cluster members to which it belongs. Then coordinate the transmission to overlapping area heads (which are shared with other neighboring clusters).

Identification of clusters

is similar to LEACH except the possibility for a node to belong to several clusters simultaneously, and reflects situations when the *received signal strength* (RSS) by the node (from distinct cluster heads) during set up is either equal or close to some extent or greater than a threshold value. The disjointness restriction between clusters is unnecessary herein. After the clusterheads are elected, the clusterheads advertise to all sensor nodes in the network that they are the new clusterheads. Once the sensor nodes receive the adverisssement, they determine the clusters they want to belong based on the received signal strength of the advertisement. Each sensor node has the possibility to join more than one cluster, and inform the appropriate clusterheads it will be member of the cluster.

Afterward, the clusterheads assign the time on which the sensor nodes can send data to the clusterheads based on TDMA approach.

Identification of overlapping area heads (OAHs)

Every node, which is belonging to more than a cluster is a potential candidate overlapping area head(OAH), thus it must inform the corresponding cluster head of every cluster to which it belongs to. Preference should be given to those candidates which are associated with the maximum number of clusters. Once a candidate is elected it becomes an overlapping area head(OAH), typically a CH may have several OCHs associated with it. Ideally one cluster should have a common overlapping area head with each of the neighboring clusters. However, this is not required as it seems that it is not always the case in practice due to the random deployment of sensors over the field area. Basically, an overlapping area head ensures inter-cluster communication between the associated pairs of clusters to which it belongs.

III.3.2 .2. *The steady phase:*

The steady phase is executed in several iterations. Role of the clusterhead(CH): firstly using TDMA the CH(of each cluster) aggregate data from all of its cluster members, then broadcast it to every overlapping area head (OAH) belonging to its cluster in one-hop. As illustrated in Fig.2 every OAH in turn using an extended version of time-division multiple access **(**TDMA) relay data the other associated CHs in other related clusters in one-hop, i.e., the OAH play the role of a CH in the newly formed cluster containing related CHs. Then again the CHs broadcast aggregated data (already obtained from all of its related OAHs) to OAHs using extended TDMA. The algorithm proceeds iteratively in a similar way, while the data propagates progressively until reaching the sink. More specifically, this communication scheme is implemented using an extended version of time-division multiple access **(**TDMA) allowing the allocation of several slots for a given node, and alternating the role of the clusterhead and its associated overlapping area heads. Put it simply, an OAH play the role of a cluster head of the cluster formed by the CHs associated with the clusters to which it belongs. MRL denotes the maximal number of the radio links that an OAH might have to establish with CHs. We used the following heuristic: for each OAH and for each two associated clusters, we elect the OAH with the maximum MRL.

After a certain period of time spent on the steady phase, the network goes into the set up phase again enter another round of electing clusterheads and overlapping area heads.

It is worth mentioning that while CHs are elected randomly, OAHs are selected heuristically. Errors caused by failing hardware or drained batteries are the norm rather than the exception in WSN. Robustness is ensured through the possibility of multiple paths to convey data to the Sink. Scalability in such a protocol scheme is ensured and in fact *built-in* due to the local decision nature of the algorithm used to implement the protocol.

III.3.3 *Assumptions, constraints and requirements*

Without any lose of generality and for the sake of practicality we made the following assumptions:

- Each node has to belong to at least one cluster to ensure the recovering of all nodes in the sensor field. A cluster may or may not be dense or sparse depending on the random deployment in the sensor field.
- A cluster should contain at least two nodes (one will be its CH and the other will be its OAH)
- Each CH belongs to only one cluster (by construction)
- Each cluster has at least one OAH, i.e. it overlaps with a least another neighbor cluster.
- Each OAH head is shared by at most k clusters, where k is a certain fraction of c the number of clusters. It is not necessary that clusters have an equal number of OAH, and depending on the random deployment of sensor in the field will likely have a different number.
- The sink is assumed to be reachable at least through either one CH node or one OAH node, which means that the sink might be thought of as a member of at least one cluster. The sink will be clustered automatically during the set up phase.

The number of sensor nodes involved in this protocol is less than the one involved in a flat multi-hop scheme. It has the advantages and features of multi-level hierarchical cluster-based scheme while it can accommodate multi-path routing by construction. More over in hierarchical scheme only cluster heads(CHs) are involved. Obviously, this protocol acts better than those protocols in terms of optimizing cluster heads as well as overlapping area heads (OAHs) energy consumption, amount of data gathered, and extending network lifetime. This protocol exhibits inherent fault tolerance and provides the possibility to save energy within the network.

IV. CONCLUDING REMARKS AND FUTURE WORK

We have described a novel multi-hop hierarchical cluster-based multi-path routing energy-aware protocol that prolong the sensor network lifetime while ensuring robustness and fault-tolerance. It takes into account the broadcast nature of radio. Moreover, this protocol is scalable to large and dense sensor networks that consist of thousands of nodes.

Of particular interest to us is studying design tradeoffs between *performance*, *survivability* and *scalability* of WSNs. *Survivability* is the degree to which essential functions of the WSN are still available even though some part of the system is down. The goal of this investigation is a high capacity WSN that degrades softly under attack, while always providing a critical level of service. Realizing the potential of large, distributed WSNs requires major advances in the theory, fundamental understanding and practice of distributed signal processing, self-organized communications, and low level as well as high level *information fusion* [13-16] in highly uncertain environments more importantly their seamless integration in a unified coherent framework.

REFERENCES

[1]Pahlavan K. and A. Levesque. 1995. *Wireless Information Networks*. John Wiley & Sons, Inc., New York.

[2]Zhao F. and L. J. Guibas. 2004. *Wireless Sensor Networks: An Information Processing Approach*. The Morgan Kaufmann Series in Networking, Elsevier, Amsterdam.

[3]Terrance J. Dishongh, M. McGrath, and B. Kuris 2009. *Wireless Sensor Networks for Healthcare Applications*, Artech house.

[4]Samuel C. Yang. 2004. *3G Wireless System Engineering, Artech house.*

[5]Akyildiz I. F. et al., "A Survey on Sensor Networks," *IEEE Communications Magazine*, August 2002, pp. 102-114.

[6]Akyildiz I. F. and C. V. Mehmet 2010. *Wireless Sensor Networks*, Wiley.

[7]Boukerche A. (Editor). 2008. *Algorithms and Protocols for Wireless Sensor Networks*, Wiley.

[8]Haykin S. and K. J. Ray Liu 2009. *Handbook on Array Processing and Sensor Networks*, Wiley.

[9]Nayak A. and Stojmenovic I. 2010. *Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication*, Wiley.

[10]IST Advisory Group. 2001. *Scenarios for Ambient Intelligence in 2010*, European Commission.

[11]Vasilakos A. and P. Witold. 2006. *Ambient Intelligence, Wireless Networking, and Ubiquitous Computing*, Artech house.

[12]Heinzelman W., A. Chandrakasan, and H. Balakrishnan. 2002 "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," *IEEE Trans. on Wireless Communications*, pp. 660–670, (Oct.)

[13]Bosse E., J. Roy, and S. Wark. 2007. *Concepts, Models, and Tools for Information Fusion*, Artech House.

[14]Hall D. L. and S.A.H. McMullen. 2004. *Mathematical Techniques in Multisensor Data Fusion,* Second Edition, Artech House.

[15]Beldjehem M**.** 2009, "A Granular Unified Framework for Learning Fuzzy Systems," in Journal *of Advanced Computational Intelligence and Intelligent Informatics* (JACIII), Vol.13 No.4, pp. 520-528.

[16]Beldjehem M**.** 2011 "On Conception and Validation of Hybrid Relational Min-Max Learners," *International Journal of Hybrid Information Technology (IJHIT)* , Vol. 4, No.3, pp. 1-21.

Appendix

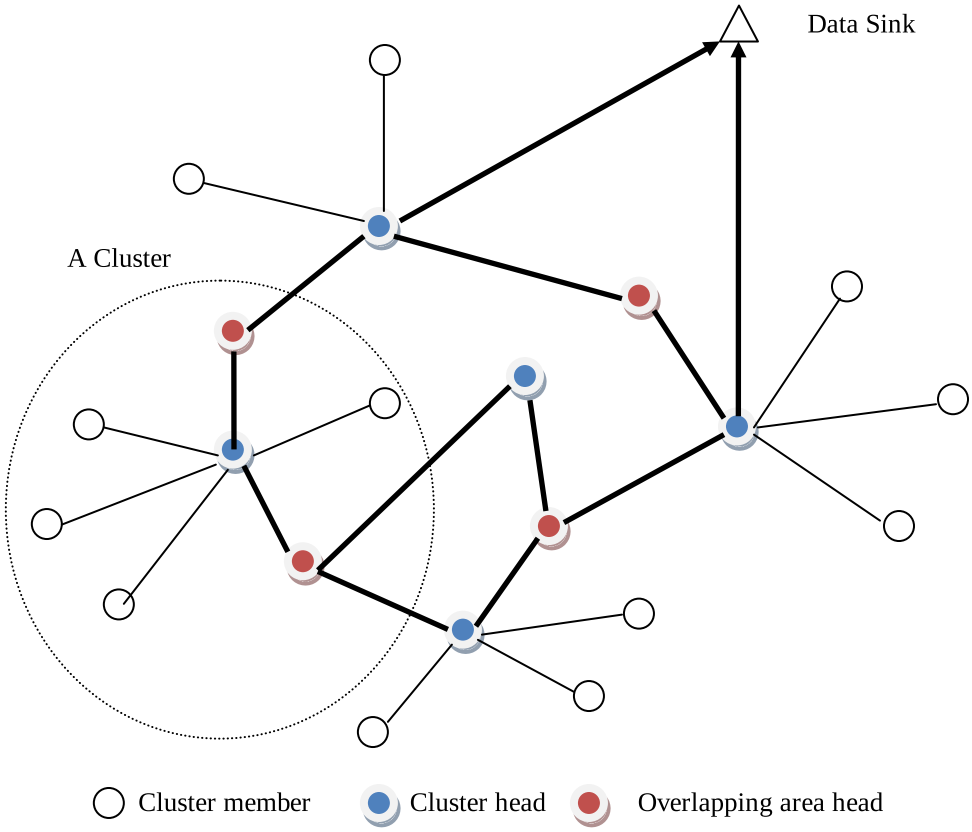


Figure 1. Network style with fuzzy soft clustering

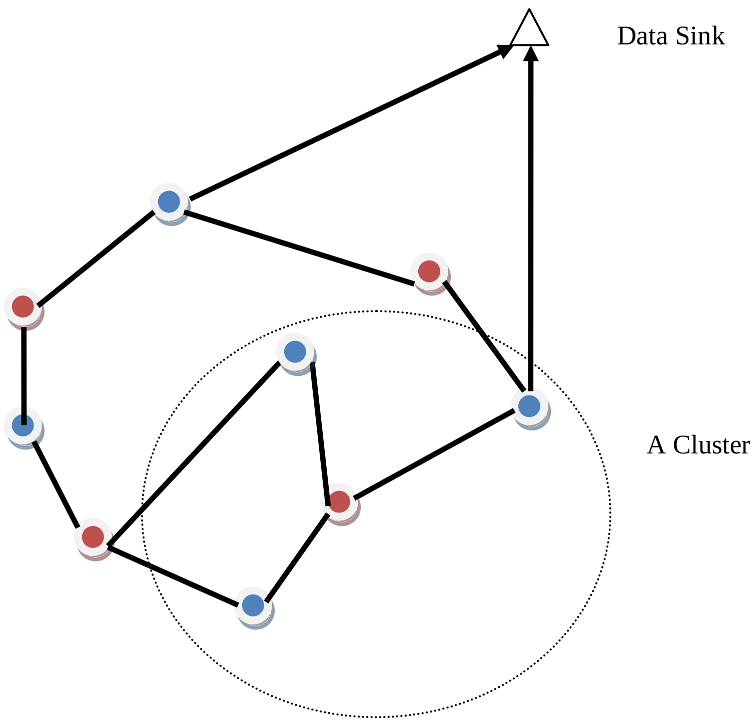


Figure 2. An OAH playing the role of a CH and CHs playing the role of cluster members during the steady phase.

CROSS PROTOCOL SIP/RTP INTRUSION DETECTION AND PREVENTION SYSTEM: RECOMMENDATION & PROTOTYPE IMPLEMENTATION

Tauqeer Safdar
Fatima Al-Abri
Department of Information Technology
Higher College of Technology, P.C. 133, P.O.Box 74, Muscat,
Sultanate of Oman
E-mail: tauqeer.safdar@hct.edu.om

KEYWORDS
Model design, Computer Networks, VoIP, Intrusion Detection & Prevention System.

ABSTRACT

In this paper, SIP/RTP Intrusion Detection and Prevention requirements are analyzed and IDS/IPS architecture is proposed using popular open-source software Snort Inline (an iptables based Intrusion Detection and Prevention System). The proposed architecture extends the basic functionality of Snort, making use of the preprocessing feature that permits analyzing SIP/RTP. For behavior based intrusion detection and prevention we have used network mode out of many available in Snort For knowledge based detection Snort Inline mode has been used with the help of iptables. The paper also includes some extensions to the basic rule language of Snort for the implementation of SIP/RTP Intrusion Detection and Prevention. The implementation is tested only on the local system with Snort Inline mode and the configuration works well.

INTRODUCTION

Voice over IP (VoIP) systems is gaining in popularity as the technology for transmitting voice traffic over IP networks. Along with the anticipated widespread adoption of VoIP systems comes the possibility of security attacks targeted against such systems. Attacks against such systems serve as the design motivation for an intrusion detection system (IDS). Indeed, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) should control both signaling (SIP) [1] and voice (RTP) [2] traffic in order to securing the network. In particular, from a security viewpoint, VoIP presents two main peculiar features: RTP communication can be routed independently of the call setup path and security should definitely be handled without sacrificing the quality of voice communication. The first characteristic implies a necessity for a deep investigation to determine the validity of the RTP packets (e.g. call setup and live communications may need to be closely observed in order to verify its legitimacy). The second feature is particular relevant since data transmission allows rather long transmission delays as long as its content is accurately analyzed. The proposed architecture is based on above mentioned two steps. The architecture is able to analyze the SIP protocol and the RTP protocol to address the interaction and the attacks dealing with cross-protocol checks. The proposed architecture based on the existing security tool by extending its functionality and its rule language extensibility for the detection and prevention of cross protocol intrusions and only configured with the Snort Inline [3] on locally system with the help of iptables [4].
The paper is organized as follows. Section II presents a brief description SIP/RTP Intrusion Detection and Prevention. Section III describes the developed Snort SIP/RTP processor prototype, while Section IV finally describes the conclusion remarks of the paper while delineating next improvements planned.

SIP/RTP INTRUSION DETECTION AND PREVENTION:

Intrusion detection, in the general sense, identifies anomalous, inappropriate, or incorrect access to a system. There has been much work on defining the types of intrusions [5, 6, 7, 8, 9], distinguishing an intrusion from normal activity [10, 11, 12], and prototyping various intrusion systems [13, 14, 15, 16, 17, 18].
There are three basic ways to detect an intrusion: anomaly detection, signature detection, and learning. The term "intrusion prevention" has caused quite a stir in the IDS community in the last few years. An intrusion prevention application is similar to IDS in that both applications aim to distinguish unauthorized activity from normal activity. An intrusion prevention application, like an IDS, has a set of signatures or predefined conditions that, when met, trigger a response. This response itself, however, differs, and is what differentiates IDS from an intrusion prevention application. Intrusion detection and prevention techniques are either based on statistical anomaly observation or on signature-based detection, also known as behavior-based and knowledge-based techniques respectively. Both techniques have pros and cons: behavior-based techniques can detect previously unknown attack methods but have a higher false-positive rate while knowledge based techniques can detect only known attacks but the false-positive rate is much lower. SIP/RTP intrusion detection and Prevention requirements chosen the Network based Intrusion Detection System for the good logging nature. In terms of scalability the Network based intrusion detection is the best choice. We regard the knowledge based techniques as the first ones to be implemented, as given the security taxonomy [19]. In our opinion the behavior based techniques should be

implemented on the top of knowledge based techniques once they have filtered out the malicious packets.
The proposed architecture is to function at the network level and we used IDS with two state checks: Knowledge based and Behavior based. We propose knowledge base intrusion detection at kernel level by using the iptables [4] with Snort Inline [3] for SIP/RTP. Iptables/Netfilter is a set of hooks inside the Linux kernel that allows kernel modules to register call back functions with the network stack. A registered callback function is then called back for every packet that traverses the respective hook within the network stack. Iptables is a generic table structure for the definition of rulesets. Each rule within an IP table consists of a number of classifiers (iptables matches) and one connected action (iptables target) [20]. This combination offers best intrusion detection and prevention when used in LAN.

RELATED WORK

The prototype implementation presented in [21] is able to analyze the SIP protocol but lacks the ability to analyze the RTP protocol. It was implemented using the Snort software. Snort is a very popular open source Network based IDS and IPS written in C language. Snort is able to capture the traffic from the Network Interface Card (NIC) using the libpcap libraries when working in the IPS mode, it is able to act as a transparent bridge between all the traffic at application level using iptables and the ip_queue module [4]. To serve this purpose the author developed a SIP preprocessor and placed it in the preprocessor block of snort.

SNORT SIP/RTP INTRUSION DETECTION SYSTEM AND PREVENTION SYSTEM

A prototype of a SIP/RTP intrusion detection and prevention system is implemented using the Snort Inline software [3] as basis. Snort is a very popular open source Network based Intrusion Detection and Prevention System written in C language. We present only the basic architecture with Figure 1. [21].

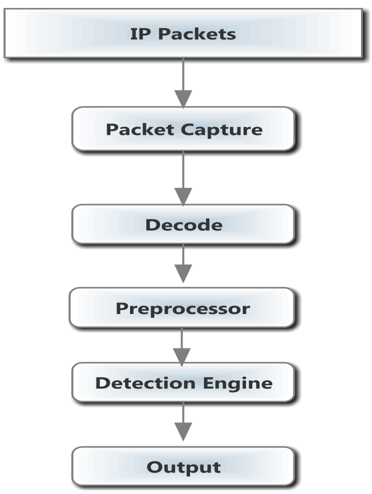


Figure 1- Block scheme of Snort architecture [21].

The detail of each block of above figure (1) is given as:
• Packet Capture block: It is the block responsible to capture the packets and pass them to following blocks, it uses iptables to prefilter the packets for the Snort Inline Preprocessor.
• Decoder block: It is the block responsible to perform the syntax analysis at layer 2, 3 and 4 of the IP packet (MAC, IP and TCP/UDP layer).
• Preprocessors block: It is the block where multiple preprocessors can be loaded at boot time to analyze protocols of layers above the TCP/UDP one with custom made C/C++ programs.
• Detection Engine block**:** It is the block where signatures and rules are checked (signature based detection and prevention) to analyze protocols of layers above the TCP/UDP one.
• Output block: It is the block managing the log output. The output log is configurable depending on user needs.
Keeping view of the above functionalities of different blocks, we used iptables in Packet Capture Block (PCB) with the help of Snort Inline mode. So when Snort works in Inline mode the Network Interface captures the packet with iptables applying knowledge base checks to adds the functionality of Prevention performing any of the following possible actions:
• Drop – the drop rule type will tell iptables to drop the packet and log it via usual Snort means.
• Reject – the reject rule type will tell iptables to drop the packet, log it via usual Snort means, and send a TCP reset if the protocol is TCP or an ICMP port unreachable if the protocol is UDP.
• Sdrop – this rule tell the iptables to drop the packet and nothing is logged.
For additional security checks for SIP and RTP packets we used the RTP proxy for analyzing the RTP ports and also the basic RTP library for its header information and payload.
In SIP preprocessor developed by the [21] only SIP packet is considered for the checks. We propose a preprocessor architecture of a SIP/RTP to detect the SIP/RTP inconsistencies for intrusion detection and prevention system. The proposed architecture of SIP/RTP preprocessor is presented in Figure 2. In the proposed block scheme a packet is passed to the SIP/RTP processor only if predefined configurable conditions are met for SIP and RTP port numbers (5060, 5061 for SIP and 5004, 5005 for RTP) otherwise the SIP/RTP preprocessor ignores the packet and it directly goes to the Detection Engine block. These port numbers are passed to the configuration file in the startup of the preprocessor file. Here if a packet belongs to either of SIP and RTP then it will be further filtered for the following security considerations.
The native rule language of Snort is not well-suited for cross protocol and stateful detection. Snort provides very limited capability for remembering state within a session for a given protocol and cross protocols. To make up for this, we added some constructs to the existing rule language so that it will better suited for detecting attacks targeted at VoIP environments that span packets in a session using different protocols. The constructs provided are listed below:
(a) var - This construct is used to set the integer value of a variable in case of a rule match. This is used as a way of keeping state. It belongs to the 'option' part of a Snort rule.
(b) Protocol-Specific Constructs - To detect attacks it is necessary to look into the specific fields in the header of a protocol. Using the allowable sequence numbers for an RTP packet, we are able to add support for the SIP and RTP

protocols in Snort with the protocols already supported (TCP, UDP, ICMP and IP).
For this we currently take the session id for parsing the SIP and the RTP port fields from the SDP header of SIP. With this the range of security checks on attacks also cover attacks aimed at RTP packets. This architecture is therefore able to perform SIP/RTP knowledge base checks as well as it incorporates a basis for the behavior based checks maintaining the stateful behavior for the active transactions.

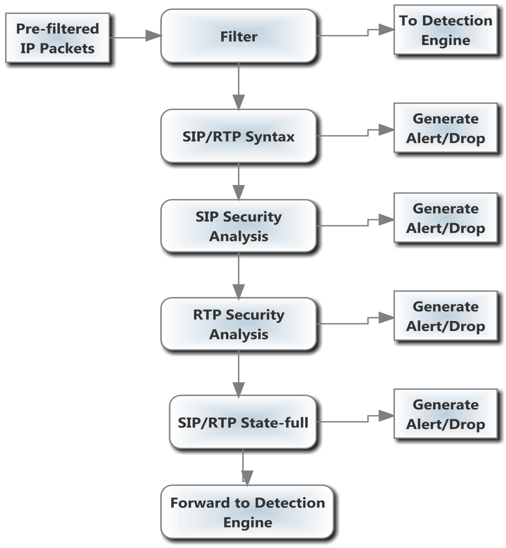


Figure 2: Block Scheme of SIP/RTP Preprocessor Logic.

CONCLUSION

We analyzed SIP/RTP Intrusion and Prevention requirements and proposed an IDS/IPS architecture specific for VoIP applications. The system is implemented in well known open source software Snort Inline. The proposed architecture includes features for VoIP security related to the call management protocol(s) (SIP) and RTP threats (e.g. call hijacking, BYE attack and DOS etc). So this architecture extends the basic functionality of Snort by implementing the SIP/RTP knowledge based security checks and provides the basis for the implementation of behavior based technique keeping with the stateful analysis. Among all the existing security solutions this is a very simple one which only extends basic function of snort and does not create overhead on the network. Our future work is focused on carrying out performance evaluation of this system o find whether this extension in functionality imposes any performance degradations practically.

REFERENCES

[1] J. Rosenberg et al., "SIP: Session Initiation Protocol", IETF RFC 3261, June 2002.
[2] H. Schulzrine et al., "RTP: A Transport Protocol for Real-Time Applications", IETF RFC 3550, July 2003.
[5] James P. Anderson. Computer security threat monitoring and surveillance. Technical Report Contract 79F26400, Box 42, Fort Washington, PA 19034, USA, 1980.
[6] Herve Debar, Marc Dacier, and Andreas Wespi. Towards a taxonomy of intrusion-detection systems. Computer Networks, April 1999.
[7] Dorothy E. Denning. An intrusion-detection model. IEEE Trans. Softw. Eng. 1987.
[8] M. Esmaili, R. Safavi-Naini, and J. Pieprzyk. Computer intrusion detection: A comparative survey. Center for Computer Security Research, University of Wollongong, May 1995.
[9] Biswanath Mukherjee, L. Todd Heberlein, and Karl N. Levitt. Network intrusion detection. In IEEE Network, May/June 1994.
[10] Stefan Axelsson. The base-rate fallacy and its implications for the difficulty of intrusion detection. In CCS '99: Proceedings of the 6th ACM Conference on Computer and Communications Security, ACM Press, 1999.
[11] Stefan Axelsson. Intrusion detection systems: A survey and taxonomy, Chalmers University of Technology, Sweden, Department of Computer Engineering, 2000.
[12] Khaled Labib. Computer security and intrusion detection. Crossroads, 2004
[13] Debra Anderson, Thane Frivold, Ann Tamaru, and Alfonso Valdes. Nextgeneration intrusion detection expert system (NIDES), Software Users Manual, Beta-Update release. Technical Report SRI, Computer Science Laboratory, SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025-3493, May 1994.
[14] H. Debar. Intrusion detection products and trends. In Proceedings of the Connect 2000.
[15] Wenke Lee and Salvatore J. Stolfo. A framework for constructing features and models for intrusion detection systems. ACM Trans. Inf. Syst. Secur, 2000.
[16] T. Lunt, A. Tamaru, F. Gilham, R. Jagannathan, P. Neumann, H. Javitz, A. Valdes, and T. Garvey (1992). A real-time intrusion detection expert system (IDES). Technical report, Computer Science Laboratory, SRI International, 1992.
[17] Michael Sebring, Eric Shellhouse, Mary Hanna, and R. Alan Whitehurst. Expert systems in intrusion detection: A case study. In Proceedings of the 11th National Computer Security Conference, National Institute of Standards and Technology/National Computer Security Center, October 1988.
[18] S. R. Snapp, J. Brentano, G. V. Dias, T. L. Goan, L. T. Heberlein, C. Ho, K. N. Levitt, B. Mukherjee, S. E. Smaha, T. Grance, D. M. Teal, and D. Mansur. DIDS (Distributed Intrusion Detection System) – In Proceedings of the 14th National Computer Security Conference, October 1991.
[19] VOIP Security Alliance (VOIPSA) Security and Privacy Threat Taxonomy, Public Release 1.0, October 2005.
[20]. Paul Russel: Netfilter Hacking HOWTO, 2005.
[21]. SIP Intrusion Detection and Prevention: Recommendations and Prototype Implementation by S. Niccolini, R. G. Garroppo, S. Giordano, G. Risi, S. Ventura in IEEE in 2006.

WEB REFERENCES

Netfilter home page, http://www.netfilter.org
Snort home page, http://www.snort.org

BIOGRAPHY

TAUQEER SAFDAR is a Lecturer in the Department of Information Technology (Networking), Higher College of Technology (HCT), Muscat, Oman. He completed his BS in Computer Science from BZU, Multan, Pakistan, MS in Computer Science specialization in Computer Networks from CIIT, Wah Cantt, Pakistan and his PhD in Information Technology (Networks) from University of Technology, PETRONAS, Tronoh, Malaysia, is in progress. His research interests include networking issues in cognitive radio networks, wireless networks, educational technology, security and reinforcement learning.

BLOCKING COMMUNICATION GRID VIA ILL-INTENTIONED TRAFFIC

Zaitsev D.A.[1], Shmeleva T.R.[2]
International Humanitarian University,
Fontanskaya Doroga, 33
Odessa 65009, Ukraine
daze@acm.org[1]
tishtri@rambler.ru[2]

Retschitzegger W.[3], Proll B.[4]
Johannes Kepler University,
Altenbergerstraße 69,
A-4040 Linz, Austria
werner.retschitzegger@jku.at[3]
birgit.proll@jku.at[4]

KEYWORDS
Computing grid, traffic, performance evaluation, colored Petri net, deadlock

ABSTRACT

A rectangular communication grid model was constructed in the form of a colored Petri net as a composition of submodes of terminal devices producing and consuming packets and communication devices switching packets for their delivery among terminal devices. The model was supplied with traffic guns simulating ill-intentioned traffic. Via simulation in CPN Tools it was shown that a traffic duel of a pair of guns brings the grid to a full deadlock with less than ten percent of the peak load. Thus, the vulnerability of grids to traffic attacks was revealed. The future work is aimed to resist the attacks.

1. INTRODUCTION

Grid and cloud computing [1,2] represents an alternative to the centralized supercomputer concept and possesses a series of advantages among which the following should be mentioned: high reliability because a failure or breakdown of a single node influences the operability of the entire grid negligibly; possibility to gather idle computing resources of the entire world; accessibility of information, applications, and processors from any access point.

Wide spreading of grid and cloud computing brings along with considerable advantages some imperfections caused by the vulnerability of rapidly grown structure. So, in [3,4], a possibility of blocking a regular grid structure via ill-intentioned traffic was revealed. Since classical Petri nets were applied, the numerical characteristics such as probability of blocking, percentage of the grid performance fall, and its influence on QoS were not investigated.

Advantages of modeling systems, and, in particular, networks via colored Petri nets (CPN) were discussed in [5,6]. CPN contain facilities for modular (hierarchical) composition of models, description of timed characteristics and statistical processing of simulation results. Colored Petri nets of simulating system CPN Tools were applied successfully to the performance and QoS evaluation of modern networking technology such as Ethernet, MPLS, PBB that is reflected in the kit of real-life examples situated on its site [7].

One of the first works on modeling grids via CPN [8] is devoted to the grid workflows; in [9] special kind of prediction Petri net was introduced; in [10] Petri nets describe coordinated cyber-attacks on grids.

The present work elaborates Petri net grid models [3] bringing them to the class of CPN with the goal of the numerical evaluation of consequences, which ill-intentioned traffic, appearing a threat to the grid operability, affects. As a future research direction, counter measures to resist this threat are planned.

2. COLORED PETRI NET MODEL OF GRID

A colored Petri net, according to [5], represents a Petri net graph whose elements are loaded with a functional programming language CPN ML. Petri net graph consists of places, depicted as ovals, and transitions, depicted as rectangles, connected via arcs. Dynamical elements, named tokens, are situated inside places; they are consumed and produced as a result of transitions' firing. A CPN token is an object of an abstract data type named a color set. Arcs and transitions are loaded with CPN ML predicates and functions representing convenient tools for modeling.

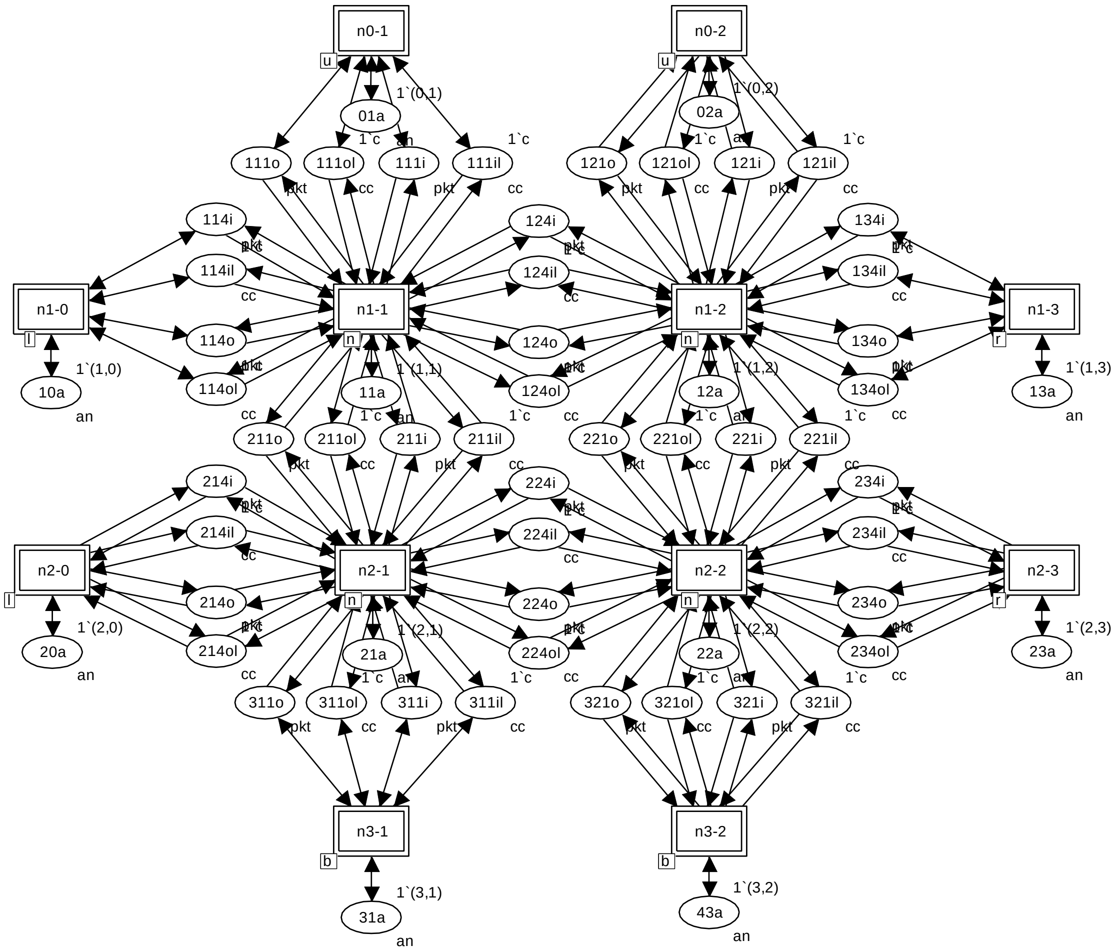
A timed delay could be associated either with a transition or with an arc; delayed tokens wait elapsing time into output places of a transition. To indicate a trace of model behavior, two magnitudes are employed: *Step* – number of fired transitions and *Time* – current value of the model time. Each timed token k has a timestamp t of its activation written in the form $k@t$; delays are expressed in the form $@+d$ that means an increment of the current model time to reach the token activation time. Such a bit complicated scheme allows transitions to fire instantaneously.

Modular composition of a model is provided via the operation of transition substitution when a transition maps to a subnet. As a subnet interface, contact places, named ports, are indicated with tags *In*, *Out*, and *I/O*. Transition substitution imply attaching a tag of subnet to the transition and mapping ports into sockets – places of higher level net incidental to the transition.

Specific issues of networks' models construction in CPN Tools are studied in [6]; CPN Tools software and example real-life models could be downloaded from [7].

2.1. Composition of Grid Model

Grid model is composed as a square (rectangular) matrix of data communication equipment (DCE) supplied with data terminal equipment (DTE) attached on the borders. Each DCE has four ports situated on the sides of a unit size square and works in full-duplex mode providing two channels for independent transmitting and receiving packets. DCE implements switching packets among ports based on the store-and-forward principle. DTE produces and consumes packets to model either grid workload or special (ill-intentioned) traffic.



Figures 1: An Example of Square Grid Model with Size 2x2.

Typical DCE model is named *n* (abbreviation of “node”). Typical DTE models are named *l, r, u, b* (abbreviations of “left”, “right”, “upper”, and “bottom” borders’ names correspondingly); they have minor difference as it is described in sequel.

We use a system of nodes’ addressing within a rectangular grid of size $k1 \times k2$ with two integer numbers *(i,j)* according to Fig. 1 where the first number denotes a row and the second – denotes a column. DCE are enumerated within the range from *1* to *k1* in vertical direction and from 1 to k2 in horizontal direction while DTE have either of indices equal to *0* or *k1+1 (k2+1)*.

Description of the node address has the following form

*colset an = product INT * INT;*

Model of a grid is composed of models of nodes; name of a node contains suffix equaling to its address; each node is supplied with a place with suffix “a” having its address. Type of node is defined by the transition substitution tag written in small rectangle beside it: *n, l, r, u, b*.

A packet is transmitted between a pair of DTE and has the following description

*colset pkt = record da:an * sa:an * co:STRING timed;*

where *da* is a destination address, *sa* is a sender address and a string *co* represents some content of a packet.

To calculate buffers’ sizes, we use elementary tokens of the following form

colset cc = unit with c;

Each channel is described by a pair of places: one of type *pkt* to store a packet and another of type *cc* to store the buffer size. Buffer sizes are measured in number of packets; port buffers have size equal to unit. Each port consists of two channels: input with suffix “i” to receive packets and output with suffix “o” to transmit packets. Thus, a port is represented by four mentioned contact places for connection with a neighbor node. Enumeration of ports is done clockwise starting from the upper port whose number is equal to unit.

Connection of nodes to assemble a grid model is implemented via merging corresponding contact places of neighbor devices as shown in Fig 1. In horizontal direction port 2 of the left node is merged with port 4 of the right node; in vertical direction port 3 of the upper node is merged with port 1 of the bottom node. To avoid duplicity in the port names, we consider only names of the left (number 4) and the upper (number 1) ports of the current node. The right (number 2) and the bottom (number 3) ports' names do not appear within the model; instead of them we use names of the left (number 4) and the upper (number 1) ports of the neighbor nodes. Thus, suffix "i/o" corresponds to the input/output channel of either the left (number 4) or the upper (number 2) port of one of the connected nodes.

2.2. Model of a DCE Node

Model of a DCE node is shown in Fig. 2. It consists of $4 \times 4 = 16$ described above contact places of ports situated on the sides of a square: *p1o, p1ol, p1i, p1il, p2o, p2ol, p2i, p2il, p3i, p3il, p3o, p3ol, p4i, p4il, p4o, p4ol*. The order of places in Fig. 2 provides the connection of channels of opposite types when composing a grid: the input channel is connected with the output channel of a neighbor node and vise versa. Port 2 is connected with port 4 of the right neighbor node and port 3 is connected with port 1 of the bottom neighbor node. Contact place *ma* contains address of the current DCE node.

The internal buffer of a node is represented by the five following places: places *pb1, pb2, pb3, pb4* store packets redirected to the corresponding port while place *pbl* represents the available buffer size. Moreover, places *pb1, pb2, pb3, pb4* are complimentary to place *pbl* that means that storing a packet in either of *pb1, pb2, pb3, pb4* takes a token from *pbl* and extracting packet from either of *pb1, pb2, pb3, pb4* puts a token into *pbl*.

The output channel of a port is modeled by a single transition – for four ports *t1o, t2o, t3o, t4o* correspondingly. For instance, for port 1, transition *t1o* takes a packet from place *pb1* and puts the packet into place *p1o*; besides *t1o* takes a token from *p1ol* and puts a token into *pbl* modeling changes in the buffers' sizes. The condition of transition *t1o* firing includes presence of a packet in the corresponding section of the internal buffer *pb1* and availability of the destination port output buffer – presence of a token in place *p1ol*.

The input channel of a port is modeled by three transitions – a transition for each possible direction of transmission, excluding the packet arrival port; thus the packet redirection decision is modeled. For instance, for port 1, transition *t1i2* models redirection to port 2, *t1i3* – to port 3, *t1i4* – to port 4. Configuration of transition incidental arcs differs by the internal buffer section only. For instance, transition *t1i2* takes a packet from *p1i* and puts the packet into *pb2*; besides *t1i2* takes a token from *pbl* and puts a token into *p1il* modeling changes in the buffers' sizes. The condition of transition *t1i2* firing includes presence of a packet in the port 1 input buffer *p1i* and availability of the internal buffer – presence of a token in place *pbl*.

Switching packets at random is the simplest solution but it does not provide correct packets' delivery according to the destination address. Within the model, the packet switching decision is implemented via four redirection predicates *to1(p,a), to2(p,a), to3(p,a), to4(p,a)*. They are used as guard functions of transitions redirecting a packet to the corresponding port. A pair of predicates can be true for a given packet; in this case the direction choice is done at random among corresponding transitions. The choice could be deterministic, for instance, when a direction with greater difference of addresses is preferred but random choice works better.

The switching algorithm is based on the valid direction of transmission defined by the difference of the current and destination address on horizontal and vertical coordinate axis given by predicates *v*. It's the main rule with exceptions for the borders of the communication grid. When destination DTE and current DCE are on the same border ($db \wedge cb$), the packet should not been delivered to a wrong DTE. The application of the main rule allows transmission in both permitted directions but DTE does not provide packets redirection; thus, the direction to the DTE different from the destination DTE should be invalid. The special case ($db \wedge cb \wedge nb$) represents the direction to the neighbor destination DTE which is permitted.

Redirection predicates use the following auxiliary predicates, where constants *k1* and *k2* define the size of the rectangular grid in the vertical and horizontal directions correspondingly.

<u>Valid direction of transmission:</u>

fun v1(p:pkt,a:an)=((#1(#da p))<(#1 a));
fun v3(p:pkt,a:an)=((#1(#da p))>(#1 a));
fun v4(p:pkt,a:an)=((#2(#da p))<(#2 a));
fun v2(p:pkt,a:an)=((#2(#da p))>(#2 a));

<u>Belonging of the destination address to the corresponding border:</u>

fun db4(p:pkt)=((#2(#da p))=0);
fun db2(p:pkt)=((#2(#da p))=(k2+1));
fun db1(p:pkt)=((#1(#da p))=0);
fun db3(p:pkt)=((#1(#da p))=(k1+1));

<u>Belonging of the current DCE to the corresponding border:</u>

fun cb4(a:an)=((#2 a)=1);
fun cb2(a:an)=((#2 a)=k2);
fun cb1(a:an)=((#1 a)=1);
fun cb3(a:an)=((#1 a)=k1);

<u>Destination of the packet to the corresponding neighbor node:</u>

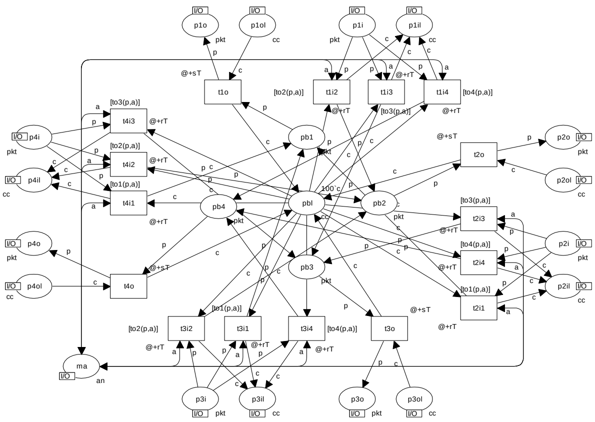
fun nb4(p:pkt,a:an)=((#2 a)=(#2 (#da p)+1)) andalso ((#1 a)=(#1 (#da p)));
fun nb2(p:pkt,a:an)=((#2 a)=(#2 (#da p)-1)) andalso ((#1 a)=(#1 (#da p)));
fun nb1(p:pkt,a:an)=((#1 a)=(#1 (#da p)+1)) andalso ((#2 a)=(#2 (#da p)));
fun nb3(p:pkt,a:an)=((#1 a)=(#1 (#da p)-1)) andalso ((#2 a)=(#2 (#da p)));

<u>The redirection predicates:</u>

fun to1(p:pkt,a:an)=v1(p,a) andalso
((not (db1(p) andalso cb1(a))) orelse (db1(p) andalso cb1(a) andalso nb1(p,a)));
fun to3(p:pkt,a:an)=v3(p,a) andalso

((not (db3(p) andalso cb3(a))) orelse (db3(p) andalso cb3(a) andalso nb3(p,a)));
fun to4(p:pkt,a:an)=v4(p,a) andalso
((not (db4(p) andalso cb4(a))) orelse (db4(p) andalso cb4(a) andalso nb4(p,a)));
fun to2(p:pkt,a:an)=v2(p,a) andalso
((not (db2(p) andalso cb2(a))) orelse (db2(p) andalso cb2(a) andalso nb2(p,a)));

The above predicates represent a result of the model behavior analysis and reflect a certain balance of simplicity with considering local information only and rather good grid performance. They themselves could be studied among other factors influencing grid performance and QoS.



Figures 2: Model of a DCE Node.

Timed characteristics of the model are given by two parameters *sT* and *rT* which represent a timed delay of sending and receiving a packet correspondingly. Remind that according to the time concept of CPN Tools, a transition fires instantaneously but the corresponding timed delay is applied to the output tokens which spend the delay into output places staying in unavailable state.

2.3. Model of a DTE Node

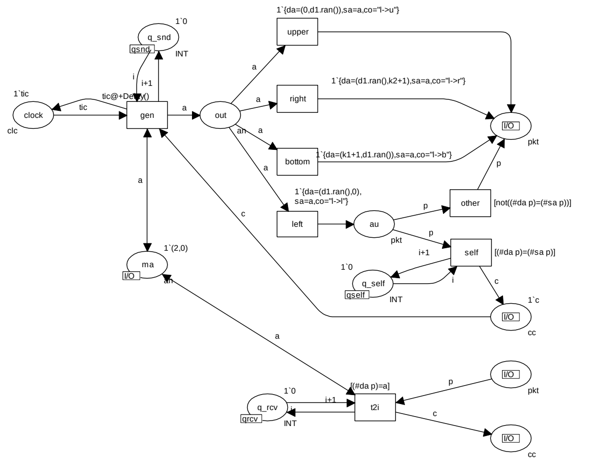
The simplified function of a DTE node is to produce and consume packets; computational aspects of DTE work could be modeled as well but they are left beyond the scope of the present paper. DTE model of a left border node is represented in Fig. 3; it is subdivided into the sending and receiving channels (tracts).

The receiving channel is modeled via transition *t2i* that consumes received packets counting their total number for the entire model in place *q_rcv* of fusion set *qrcv*. For more precise estimations, received and sent packets could be counted separately for each pair of $(2\cdot k1+2\cdot k2)^2-(2\cdot k1+2\cdot k2)$ communicating devices.

The sending channel consists of a timer represented via transition *gen* whose periodicity is controlled by place *clock*; its only token of type *tic* is delayed via random function *Delay()* which defines the period. As a result, a token, equal to the current node address *a*, is put into place *out*; it is a workpiece to produce a packet.

Either of alternative transitions *left*, *upper*, *right*, *bottom*, which are chosen on random, fires generating a packet directed to the corresponding border; its destination address is chosen on random within the border address range via standard function *ran()*; auxiliary types *d1* and *d2* have the following description

colset d1=int with 1..k1;
colset d2=int with 1..k2;



Figures 3: Model of a DTE Node (left border).

The only difference of four types of the border nodes *l, u, r, b* consists in filtering its own destination address which is done after the corresponding transition *left, upper, right, bottom*. In Fig. 3, a model of a left border node is represented, so the filtering is done after transition *left* that puts generated packet into intermediate place *au*. Transition *self* extracts and consumes packets directed to the current device counting their number for the entire model in place *q_self* of fusion set *qself* while transition *other* puts other packets to the output channel buffer of the device.

For random choice of alternative transitions and for implementation of standard function *ran()*, CPN Tools uses the uniform distribution. Besides, CPN Tools offers a wide range of known distributions to describe user random functions. For, instance, a Poisson distribution is chosen for function *Delay()* and the corresponding description has the following form

fun Delay()=poisson(10.0);

3. SIMULATING GRID WORKLOAD

Simulating the grid workload, produced by DTE models *l, u, r, b*, allows debugging the entire model, estimating the influence of its parameters on the grid performance, and studying the grid behavior under the peak load.

Basic parameters of the model, whose influence on the grid behavior was studied, are:

- DCE internal buffer size *bs*;
- intensity of the workload – a parameter of DTE model Poisson distribution *wl*;
- performance of DCE – timed delays of *rT* and *sT* of packet receiving and sending.

At small values of the internal buffer size *bs*, for instance, equal to 10, the grid falls into a deadlock even at traffic equal to 10% of the grid bandwidth. Rather bulky buffer, for instance, of size 10K and more does not allow observing a deadlock even at the peak load. In majority of simulations, the buffer size equal to 100 was chosen that allows observing either deadlocks or their absence on feasible intervals of time.

Table 1 shows that the grid is brought to a deadlock even via regular workload when the buffer size equals to 100. In majority of cases, a deadlock means that buffers of some DCE are full and the difference of the numbers of the sent and received packets is approaching the overall grid buffer size *k1*k2*bs* that equals to 6400 for Table 1 working parameters.

Table 1: Bringing the Grid to a Full Deadlock via Workload.

Workload intensity (*wl*)	*Step*	*Time*	Number of sent packets *qsnd-qself*	Number of received packets *qrcv*	Grid performance *gp* (packets/MTU)
50.0	10000000	854654	529965	529918	0,62
30.0	10000000	512757	529899	529825	1,03
20.0	10000000	341850	530026	529900	1,55
10.0	10000000	170992	530350	547421	3,1
9.0	120000000	1845409	6357110	6356608	3,44
8.0	607161*	12368	34817	29897	2,42
7.0	336234*	6413	20163	15584	2,43
6.0	200348*	2963	12659	9123	3,07
5.0	142266*	1876	8499	6261	3,34
4.0	99624*	1233	7219	3925	3,18
3.0	78952*	877	6233	2621	2,99
2.0	64852*	765	5668	1596	2,09

rT=sT=5, bs=100, k1=8, k2=8;
* *– the grid comes to a full deadlock – no permitted transitions.*

Thus, the peak load of the grid at *rT=sT=5* is about *gp=3* packets/MTU that is reached when the workload intensity is about *wl=10.0*. The abbreviation MTU means “model time unit” whose value could be chosen when scaling timed characteristics of a real-life network [6]. Note that, at a boundary workload *wl=9.*0, more prolonged simulation was done but a deadlock was not reached. When approaching a full deadlock, the grid performance can decrease in spite of high intensity of traffic because of partial intermediate deadlocks slowing down the packets delivery. For further investigation of the grid behavior, the workload of 30% is chosen that is achieved at *wl=30.0*.

4. SIMULATING ILL-INTENTIONED TRAFFIC

To simulate ill-intentioned traffic, a model of packets’ gun is constructed and its copies are connected to the grid borders. The parameters, whose influence on the model behavior is estimated, are the number and location of guns, their targets, and intensity of guns’ work.

A simple model of a packets’ gun *g* is shown in Fig. 4. Its work resembles the sending channel of DTE (Fig. 3) but both, source *a* and destination (target) *ta* addresses are given by the marking of external places *ga* and *ta* correspondingly. The periodicity of its shots is given by the random function *gDelay()* which distribution is Poisson also. The number of generated packets (shots fired) is counted in place *q_sndg* of fusion set *qsndg*. Note that, consuming of tokens, generated by a gun, is provided by the regular DTE models.

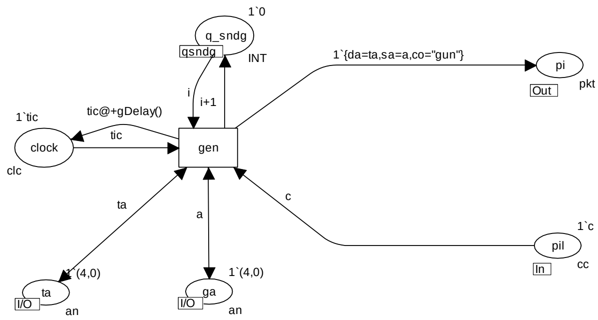
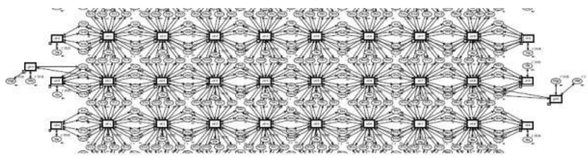


Figure 4: Model of Packets’ Gun.

The regular grid model was supplied with a few packet guns, and their influence on the model behavior was studied. As the main characteristic, the time interval for bringing the model to a full deadlock was considered. Attaching a gun or a few guns with different targets influenced the model behavior insignificantly. The most significant result was achieved via attaching a pair of guns with mutual targets – a traffic duel. An example of a traffic duel is shown on a fragment of the grid model in Fig. 5; it was denoted as *(4,0)<->(4,9)* indicating the nodes of guns’ attachment and their targets. The detailed characteristics of simulation are put in Table 2. Thus, a traffic duel causes a deadlock with an extra load of about 10%.



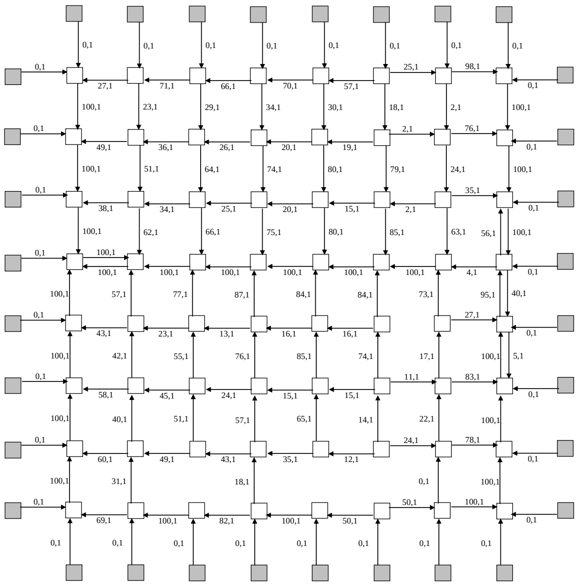
Figures 5: A Traffic Duel *(4,0)<->(4,9)* on a Grid Fragment.

Table 2: Bringing the Grid to a Full Deadlock via the Traffic Duel.

Traffic guns’ intensity (*wl*)	*Step*	*Time*	Number of sent packets *qsnd-qself*	Number of received packets *qrcv*	Grid performance *gp* (packets/MTU)
10.0	20000000	865475	1067732	1067631	1,24
6.0	20000000	784017	1071955	1071786	1,36
5.0	255240*	26556	18153	11609	0,43
4.0	188828*	23432	14515	7971	0,34

rT=sT=5, bs=100, wl=30.0, k1=8, k2=8;
* *– the grid comes to a full deadlock – no permitted transitions.*

Various other locations of guns did not produce deadlocks faster than the above studied median except of its vertical orientation. For instance, a diagonal traffic duel (1,0)<->(8,9) requires on average 8% more time to fall in a full deadlock. An example of a full deadlock is shown in Fig. 6, where inscriptions on the arcs indicate the number of packets in the corresponding section of the internal buffer and, after comma, the number of packets in the port buffer.



Figures 6: An Example of a Full Deadlock *(bs=100, wl=10.0, gl=5.0, td=(4,0)<->(4,9), Step=290529, Time=10941, k1=8, k2=8).*

Thus, simulation results acknowledged the hypothesis advanced in [3,4] that a grid could be blocked via ill-intentioned traffic. In the simplest case, two traffic generators are required ensuing a traffic duel with load about 10% of the grid peak load.

5. CONCLUSIONS

Issues of the grid blocking via ill-intentioned traffic were investigated using simulation in the environment of modeling system CPN Tools. It was shown that even at rather low workload of the grid about 30%, a grid can be entirely blocked with a full deadlock of DCE via a traffic duel with additional load little than 10%. Thus, the vulnerability of the grid structures to the attacks via traffic was revealed.

A future research direction is bringing more realism to the model and investigation of modern architectural solutions for DCE such as cut-through principle of packets' switching. Real-life devices overcome deadlocks, either local or global, using two rather simple features: they drop packets entering a busy device and clean flooded buffers tracking packets' ageing time. Thus the described deadlocks last for a short interval of time representing a cause of the performance fall.

The ultimate goal of the future work is counter measures to detect and resist traffic attacks on grids avoiding significant performance fall.

REFERENCES

1. *Grid and Cloud Computing: Concepts, Methodologies, Tools and Applications* (4 Vol.), Information Resources Management Association (USA), IGI-Global, 2012, 2134 pp.
2. Preve, N.P. (Ed.). *Grid Computing: Towards a Global Interconnected Infrastructure*. Springer, 2011, 312 p.
3. Shmeleva T.R., Zaitsev D.A., Zaitsev I.D. Analysis of Square Communication Grids via Infinite Petri Nets. *Transactions of Odessa National Academy of Telecommunication*, no. 1 (2009) 27-35. http://archive.nbuv.gov.ua/portal/natural/Nponaz/2009_1/files/20091_SmelevaZaycevZaycev.pdf
4. Zaitsev D.A., Shmeleva T.R. Verification of hypercube communication structures via parametric Petri nets. *Cybernetics and Systems Analysis*, Volume 46, Number 1 (2010), 105-114, doi: 10.1007/s10559-010-9189-y
5. Jensen, K., Kristensen, L.M. *Coloured Petri Nets: Modelling and Validation of Concurrent Systems*. Springer, 2009, 384p.
6. Zaitsev, D.A. *Clans of Petri Nets: Verification of protocols and performance evaluation of networks*. LAP LAMBERT Academic Publishing, 2013, 292 p.
7. http://cpntools.org
8. Bratosin, C.C., Aalst, W.M.P. van der, Sidorova, N.. Modeling grid workflows with colored Petri nets. In K. Jensen (Ed.), *Proceedings of the 8th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools*, 22-24 October 2007, Aarhus, Denmark, pp. 67-86.
9. Retschitzegger W., Baumgartner N., Gottesheim W., Mitsch S., Schwinger W. Situation Prediction Nets – Playing the Token Game for Ontology-Driven Situation Awareness. *Proceedings of 29th International Conference on Conceptual Modeling (ER)*, Vancouver, Nov. 2010.
10. Chen, T.M., Sanchez-Aarnoutse, J.C., Buford, J. Petri Net Modeling of Cyber-Physical Attacks on Smart Grid. *IEEE Transactions on Smart Grid*, Vol. 2 , No. 4, 2011, 741 – 749.

ENGINEERING SIMULATION

AN AGENT BASED FREIGHT ACTIVITIES AND LOGISTIC CHAINS OPTIMIZED NETWORK SIMULATOR (FALCONS)

Omar Abed
Davy Janssens
Geert Wets
Gerrit K. Janssens[b]
Tom Bellemans
Transportation Research Institute - IMOB
Wetenschapspark 5 bus 6
B-3590 Diepenbeek, Belgium
[b] Faculty of Business Economics
Hasselt University
B-3590 Diepenbeek, Belgium
E-mail: Tom.Bellemans@uhasselt.be

KEYWORDS
Freight transport, Agent based, freight model, transport policy, node capacity.

ABSTRACT

Freight movements are a result of complex and diverse logistics decisions. One main objective of recent freight and logistics models is the ability to catch and represent some of the decisions made by the different actors throughout the freight generation and distribution process. Several efforts stopped shy at the stage of providing a conceptual framework of a freight or logistics model. This is mainly due to the lack of micro level data needed to make such models operational. Few operational agent based (AB) freight models were reported where such data could be obtained. In this paper we present a conceptual framework of an agent based freight model for Flanders. The model is in early implementation stage and is planned to be operational in the foreseeable future. Micro level data at hand will make the model a capacity and policy sensitive tool to serve a range of commercial and public planning bodies. It will be shown that using AB simulation paradigm combined with data availability enabled us to capture a wide range of decisions affecting goods generation and distribution.

INTRODUCTION

Modeling freight movements have moved in recent years from a traditional four steps models based simulation towards a more micro level, activity based simulation approach. This is mainly due to availability of micro level data, computational power and new simulation techniques, like the AB simulation paradigm (Wisetjindawat et. al. 2007; Samimi et. al. 2009; Harland et. al. 2005).

Agent based simulation techniques are suitable for simulating freight movements for the following reason. Freight movements are a result of complicated chains of actions, reactions and decisions made by different actors throughout the supply chain. Decisions about route choice, mode choice and shipment size for example directly affects final total flows when aggregated. Such decisions constituting firms " behavior " take place at a micro level, matching the intrinsic nature of an agent, where fully and\or semi-autonomous agents are used to capture and represent such behaviors. Moreover, in micro level agents interactions, knowledge about the whole environment is distributed. In most advanced agent based simulation platforms, agents architecture, messaging schemes and roles can mimic some real life reactions in such an incomplete knowledge of environment. Examples of such is, a certain transport carrier firm (agent) can cease to exist if not contracted for a pre-specified period of time. The possibility to let different agent interactions evolve in time dimension and learn from history, is another reason of why AB simulation techniques are suitable in modeling freight transport chains (Davidsson et. al. 2008, Roorda et. al. 2010 ; Cavalcante and Roorda 2013).

Till now, freight flows simulation for Flanders had some limitations. The process relies on aggregate and outdated data about total flows of different goods categories. Moreover, to make future predictions, those total flows are tuned by altering Gross Domestic Product GDP indicator. Such a process is not very accurate, especially in not heavily industrialized economies. Which is the case with Flanders region, where in fact transit flows represent a significant part of freight flows across its borders. The aggregate flows of goods are then disaggregated with the ADA model (Ben-Akiva and De Jong 2013); a logistics model incorporating some logistics behaviors like shipment size and frequency and use of distribution and consolidation centers. The model takes the aggregate flows and disaggregated then to a firm to firm interaction level based on a global optimized cost function. Despite including logistics behavior elements, representation of firms population and sensitivity to link \ node capacity could be improved.

FALCONS is a freight activities and logistics chains optimized network simulator. It is an agent based model under development using the JANUS platform (Gaud et. al. 2008). Using AB simulation approach and recent availability of micro-level firm data , the model aims at a more optimized simulation of freight flows. Using real production, consumption and carrier firms data, FALCONS uses a bottom up approach in simulating freights. The estimated goods are directly linked to production and consumption level of firms, of which yearly financial figures are available. This additionally provides an up to date estimation of goods quantities, instead of using proxies like GDP rate.

In the following sections, a review of existing efforts in using disaggregate models, AB models will be introduced. After that, FALCONS model and data sources used will be detailed. Finally, we will conclude with final remarks and outline current status of the model and future work.

LITERATURE REVIEW

Traditionally, freight transport models have been using macro level data as input. Several such models exists, such as SAMGODS (Swahn, H. 2001), ASTRA framework (Rothengatter et. al 2000) and the Flemish Mobility Plan (Flemish Government 2013). This type of models is taking into account several aspects of welfare and macroeconomics and use results from intra-disciplinary sub models to simulate passenger and freight flows. A main problem with these models is that they do not take into account some or any logistical operations, like choice of carrier and shipment size, and therefore do not include the actions that lead to the size and distribution of goods flows. Models that take some logistical aspects into account are for example GoodTrip (Boerkamps et. al. 2007), SMILE (Tavasszy et. al. 1998) and ADA (Ben-Akiva and De Jong 2013). These models however - despite being more microscopic than previous ones - do not include some of the actors in the decision making process, and are blind to the time evolution dimension with which ordering and delivery processes take place.

Few AB freight or logistics models have been developed in recent years. several models are at the level of a framework or conceptual work, partially due to data unavailability. Examples of such frameworks are FREMIS (Cavalcante and Roorda 2013) which besides modeling different transportation agents, includes product differentiation and economies of scale. On the other hand, examples of operational AB models are INTERLOG (Liedtke 2009); a Germany data calibrated model, and TAPAS (Davidsson et. al. 2008); an AB transport chains simulator, used to evaluate infrastructure policy measures and business interactions. Both models captured several logistics processes, and represented most actors involved in the process of moving goods.

FALCONS will use real firm population data for the different industry sectors. Real nodes (distribution centers CCs, ports) locations and capacities. It will also use a dynamic load balancing approach to maintain node and link capacity limits. The model will also enable modelers to conduct some policy measures studies related to e.g. CO2 emissions or modal split. Or several “what if” scenarios related to firms location choice and future Production-Consumption PC scenarios, can be made using synthetic firms population generation techniques then simulated with FALCONS.

FALCONS – MODEL DESCRIPTION

In the sections following we will describe the FALCONS model, data needed to run it, agents and their interactions and expected outputs.

Introduction

FALCONS will follow a bottom up approach in modeling freight flows. Goods to be modelled follow the NST\R convention of grouping and there exists nine good categories to model. Those categories are listed in Table 1 below.

Table 1: Classification of goods categories for FALCONS

NST\R Category	Goods included
NSTR 0	Agricultural products and live animals
NSTR 1	Foodstuff and animal fodder
NSTR 2	Solid mineral fuels
NSTR 3	Petrol products
NSTR 4	Ores and metal waste
NSTR 5	Metal products
NSTR 6	Crude and manufactured minerals; building materials
NSTR 7 and 8	Fertilizers and Chemicals
NSTR 9	Machinery, transport equipment, manufactured articles…

Since the goal is to simulate freight flows for a period of a year, goods production (and hence consumption) will follow estimated time (monthly) based patterns, where certain goods will be produced and consumed more during certain times of the year. For example oil and gas production will see a small peak during winter months and so on. This time differentiation will be the entry point to the time dimension with which firms will interact with each other, and relative to which smaller time windows for ordering and delivering goods will follow.

Input and output tables (Leontif 1936) will were estimated to better match production side with consumption side, with percentage based shares as obtained from industry and literature. An example is crude oil decomposing into main final consumable products like car fuel, heating gas and pharmaceutical usage. Those approximated percentages will be used as upper limits for consumption side.

Although a complete nationwide firm database is at hand, only a sampled population of firms (representative of firm size, industry, and location) will be used. This is to overcome the computational intensive process of firm to firm interactions if all firms were modelled. Results for the sampled population will be reweighted to represent the original population. JANUS platform is able to handle approximately one million agent interactions at the messaging level used in FALCONS.

Agents in FALCONS are firms, in different sectors making a decision to deal with each other by optimizing a cost function. The cost function is a modified Economic Order Quantity (EOQ) (Muckstadt and Sapra 2010), to include further cost components as cost due to distance between agents. Cost structure for production firms is given below in (1).

$$C_{pf} = C_{shipper} + C_{inv.} \qquad (1)$$

Where, C_{inv} is the firms inventory cost per volume it possesses, and $C_{shipper}$ is cost of contract with shipping firm when moving goods from production side to target consumption side. Each cost part further breaks down to other components as shown in (2) and (3) below.

$$C_{shipper} = C_{admin} + C_{carrier} + P \qquad (2)$$

$$C_{inv.} = (I_{max} / 2)\, H + (D / Q)\, S \qquad (3)$$

Where C_{admin} is administrative cost, $C_{carrier}$ is the carrier firm cost and is a function itself of distance, fuel, vehicle maintenance, labor (driver). P is a profit margin. I_{max} is maximum inventory level, H is holding or carrying cost (Euros/ unit / year), S is ordering cost (Euros), D is demand volume per year, Q is ordered quantity.

Using data on yearly turn over for firms, and combining this with data on cost per ton per good category, we can estimate yearly firms production and consumption volumes. Those volumes will be the starting point for every firm. Volumes to send e.g. by week will be obtained by distributing those volumes over the year, following the yearly consumption patterns mentioned earlier as a guideline. This process will also serve as basis for shipment size and frequency.

Production firms will first look at possible consumption partners around them, exchange goods with them and only move to next best (less costly) consumption firm if they still possess goods to sell. After each interaction, inventories are updated to reflect current quantities, and the process is repeated till all goods at initial time are consumed. Extra quantities remaining at production side will be assumed to go for export. Unfulfilled consumption quantities from local production firms will attract import flows. Till both production and consumption sides are fulfilled.

Real carrier firms will be modeled and they will be the agents who transport goods. An algorithm to rank carrier's level of service is used where firms will rank carriers to deal with in next orders if a carrier delivers in given time window. Otherwise, firms will look for other carriers. This will reduce computational complexity arising by each firm searching for most cost efficient carrier for each time they want to send goods.

Each shipment sent will include information on source firm, destination firm, quantity, sector type and a time window to deliver. Example shipment dependencies:

shipment (source agent ID, destination agent ID, NSTR, volume, time window, mode restriction)

To reduce computational complexity and runtime, a sample of the real firms is used in every run. This still makes a large sample when firms decide to search for optimized links for transport. To reduce this complexity, the road network was skimmed to include only those roads allowed for trucks. Reducing the total amount of available road links. Additionally, some goods were restricted to certain modes as they are in real life, e.g. food related categories uses only road links and ignore rail or waterways. Finally, production agents will search for best shippers to deal with only in the first run. After that, production agents will rank this shipper as preferred choice if future deliveries need to be done to same target zone. A new search for a shipper will take place if this ranked shipper fails to deliver in time window specified.

Model agents

We will next present a description of the different agents used in FALCONS.

Firm agents

Each production, consumption and carrier firm will be acting as one agent acting and reacting to other agents following its own roles in the simulation. Each agent will have a list of attributes differentiating it from other agents, as well as a set of rules defining its interactions with other surrounding agents.

Table 2 below summarizes example attributes of a production firm.

Table 2. Example firm attributes for firm ID 203

Attribute	Value
Firm ID	203
Zone ID	127
Industry	6
Size category	2
Type (prod,cons)	2
Yearly turnover (€)	175.850,00
Yearly production size	376 tons
Special access to node	yes
Node	Antwerp port
Mode restriction	yes
Mode	rail

Firm agents are either production, consumption, shippers or carriers firms. Firms which process goods to produce other types of goods are counted twice, once as a production firm and another as a consuming firm.

Agent_production (agent ID, zone ID, NSTR, volume, inventory level, shipment_ID, time window, agent_shipper, agent_ carrier)

Carrier firms will actually handle the transport and there exist carrier firms for the three modes of interest, road, rail and waterways carriers. Each will possess a fleet of vehicles

(trucks, freight wagons or barges). Example carrier agent dependencies:

Agent_carrier (zone ID, carrier ID, mode, NSTR, # vehicles, vehicle_ID, vehicle_ID capacity, time windows, agent_shipper_ID, agent_production_ID, shipment_ID, agent_capacity_ID)

Last firm agent is shippers agents. Those agents represent shipping firms who will act as contractors of carrying services. Shippers will offer contracts to production firms in order to transport their goods to consumer firms. Shipper agents will make decisions of carrier choice, route choice, mode choice. They send their offers to productions firms based on optimized cost. They also must meet any policy constraint, coordinate with nodes (ports, DCs..) to meet delivery time windows.

Capacity agents

Capacity agents will be present in every zone and they will coordinate capacity of links and nodes usage with shipper and carrier agents. In case of restricted access to a node due to full usage of capacity at a time, capacity agents will ask carrier agents to choose next best route. Priority is given to delivery on time over cost. Example dependencies of capacity agents:

Agent_capacity (zone ID, road links, capacity road link, rail link, capacity rail link, # nodes, node type, capacity per node type, agent_shipper, agent_carrier)

Timing agent

This agent will serve as a universal clock for the simulation and make sure time windows are coordinated.

Example interaction flow

Below we illustrate a simplified example scenario of moving goods from one production site to a consumption site.

1. Agent_production (2, 127, 2, 100 tons, 600 tons, slot 3, shipper = unknown, carrier = unknown), picks (based on closest distance) agent_consumption (67, 127, 2, - 250 tons, - 670 tons, slot 3).

2. Agent_production (2, 127, 2, 100 tons, 600 tons, slot 3, shipper = unknown, carrier = unknown) contacts all shippers in its own zone ID (assumption made for simplicity) and asks for best offers to move his goods to agent_consumption (67, 127, 2, - 250 tons, - 670 tons, slot 3).

3. Every agent_shipper will contact pool of available and permissible agent_carriers in same zone first, and calculate cost of moving goods with different carriers.

4. Agent_production (2, 127, 2, 100 tons, 600 tons, slot 3, shipper = unknown, carrier = unknown), will receive offers and ranks best 3. He picks best one to use for now. Remaining 2 are preferred choices in next assignments if choice one is not available.

5. Agent_production (2, 127, 2, 100 tons, 600 tons, slot 3, 5, 35) updates his shipper and carrier values after he picks agent_shipper_5, who subcontracted *Agent_carrier* (127, 35, 1, 2, 5 heavy trucks, vehicle_2, vehicle_2 capacity = 50 tons, vehicle_3, vehicle_3 capacity = 50 tons, slot 3, agent_shipper_5, agent_production_2,shipment_1, agent_capacity_2).

6. Agent_capacity_2 checks that moving this shipment doesn't cause excessing link capacity chose by agent_carrier_35. If yes, agent_carrier_35 will chose next available link maintaining the time window for shipment delivery. If not possible, it will use original link, but agent_capacity_2 will mark the link as over congested. Consequently agent_production_2 will downgrade ranking of agent_shipper_5.

7. Otherwise, Agent_production (2, 127, 2, 50 tons, 500 tons, slot 2, shipper = unknown, carrier = unknown) updates his inventory to reflect new levels and is ready for next shipment, 50 tons in this case. Agent_consumption_67 will update his inventories too and gets ready for next shipment.

8. Process is repeated till all consumption volumes (negative inventories are filled).

Data sources and zoning system

Below we summarize data sources needed to run FALCONS and the zoning system used.

Bel-First firm database

This is a firm database for Belgium and Luxemburg updated yearly by Bureau Van Dijk (Bureau Van Dijk). Firms can be filtered out using several criteria. Attributes of interest for us and used in this work are number of firms, industry classification, address information, size (C1: very large, C2: large, C3: medium or C4: small), yearly turn over for last year. As explained earlier, yearly turnover will be used for estimating firm production volume, from which shipment size and frequency are obtained.

Zoning system

The present-day Flemish Region covers roughly 13,522 km^2 and is divided into five provinces, 22 districts and 308 municipalities. Flanders is divided hence into 308 zones, each zone representing one municipality.

Network database

This is a database containing travel time data for road links. Data include free floating (no traffic) travel time, loaded network, AM and PM peak travel times and travel distances respectively. Work is in process to enhance existing rail link database. The database also includes information on nodes (ports, DCs, and CCs).

Other data sources

Few other data files are used to run the model. Examples are value to weight data describing monetary value per ton of each good categories. Data on vehicle types, dimensions, fuel and labor cost, etc.

Discussion

Since we are using yearly turnover data to estimate volumes sold, and since this data is available for the previous year, this means the simulation process is modeling a push transport rather than pull. This means estimated volumes at start must be consumed totally, as we know they did indeed got sold and hence transported. For the purpose of modeling yearly flows, this is an acceptable assumption. However, future work on the model will aim at incorporating additional logistics practices as produce on demand.

The model focuses on internal freight flows and use remaining production volumes or unfulfilled consumption volumes as basis to estimate import and export flows. However, transit flows are still not modelled due to the very complex nature of those flows. Some representation of transit flows might be achieved if one assumed if usually nodes are operating close to their maximum capacity, and any free capacity after all internal, import and export operations are complete, will attract transit flows.

SUMMARY

We have presented FALCONS; an agent based freight model for Flanders. The aim of the model is to simulate the annual freight flows in three modes of transport for now (road, rail, inland waterways), moving between 308 geographical zones. The simulation will allow several policy measures assessment. FALCONS uses agent based simulation technique to model in a bottom up approach the different actors directly related to the move of flows, namely; production and consumption firms, carriers and logistics nodes. Recent availability of up to date and micro level data will be used to run the model. The core of the model is a cost optimizing process that the agents will use as a basis to interact with one another. The model also will use a dynamic load balancing technique to maintain capacity limits on links and nodes. The model is at early implementation state, and a working prototype is planned before the end of 2014.

REFERENCES

Ben-Akiva, M.E. and de Jong G.C. 2013. “ The aggregate-disaggregate-aggregate (ADA) freight model system “, in *Freight Transport Modelling*, Emerald Group Publishing, ISBN 1781902860, 9781781902868.

Boerkamps J. H. K., van Binsbergen A. J., Bovy P. H. L.. 2007. “Modeling Behavioral Aspects of Urban Freight Movement in Supply Chains”. *Transportation Research Record: Journal of the Transportation Research Board*. ISSN 0361-1981.Volume 1725 / 2000. Pages 17-25.

Cavalcante R., Roorda M. 2013. “Freight Market Interactions Simulation [FREMIS]: An Agent - based Modeling Framework” . *In Procedia Computer Science*, 19, , 867 – 873

Davidsson P, Holmgren J., Persson Jan A., Ramstedt L., 2008, “Multi Agent Based Simulation of Transport Chains”, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., Estoril, Portugal, pp.1153-1160.

Flemish Government . 2013. “Flanders Mobility Master Plan - Directing Part”. Department of MOW, Brussels, Belgium, p. 81.

Gaud N., Galland S., Hilaire V., Koukam A. 2008. “An Organisational Platform for Holonic and Multiagent Systems”. *Sixth International Workshop on Programming Multi-Agent Systems (PROMAS'08).* E. Hindriks, A. Pokahr and S. Sardina (Eds.), pp. 111–126. Estoril, Portugal - May 12th to 16th,

Harland K., Heppenstall A., Smith D., Birkin M. 2005. “ Creating Realistic Synthetic Populations at Varying Spatial Scales: A Comparative Critique of Population Synthesis Techniques ”. *Journal of Artificial Societies and Social Simulation*, 15, 1-1.

Leontif W.1936. “ Quantitative Input and Output Relations in the Economic Systems of the United States ”. *The Review of Economics and Statistics*, Vol.18, No.3, Aug.,1936, pp.105-125.

Liedtke G. 2009. “ Principles of micro-behavior commodity transport modeling ”. *Transport Research Part E: Logistics and Transport Review*. 45: 795-809.

Muckstadt J. A., Sapra A. 2010. “ Principles of Inventory Management ”. Springer Series in Operations Research and Financial Engineering. ISBN 978-0-387-68948-7. XVIII, 339 p. pages 17-42.

Roorda M. , R. Cavalcante, S. McCabe, H. Kwan. 2010.“A conceptual framework for agent-based modelling of logistics services”. In *Transportation Research Part E* ,46, 18– 31.

Rothengatter W., Schade W., Martino A., Roda M., Davies A., Devereux L. , Williams I. 2000. “ASTRA Methodology”. Project No: ST-97-SC.1049.Commissioned by: European Commission - Fourth Research Framework - DG VII.

Samimi A., A. Mohammadian, K. Kawamura. 2009, “Behavioural freight movement modeling”, Presented at *The 12th International Conference on Travel Behaviour Research*, Jaipur, Rajasthan, India, ,13-18.

Swahn, H. 2001. “ The Swedish model systems for goods transport – SAMGODS. A brief introductory overview”. SAMPLAN Report 2001:1, SIKA.

Tavasszy L.A., Van de Vlist M., Ruijgrok C.,Van de Rest J. .1998. “ Scenario-wise analysis of transport and logistic systems with a SMILE”. Paper presented at the *8th WCTR conference* , Antwerp, Belgium.

Wisetjindawat W., K. Sano, S. Matsumoto , P. Raothanachonkun. 2007. “Micro-simulation model for modeling freight agents interactions in urban freight movement”. In CD-ROM Proceedings, *86th Annual Meeting of the Transportation Research Board*, Washington DC.

WEB REFERENCES

Bureau Van Dijk. Company information and business intelligence. Bel-first firm database for Belgium and Luxemburg. http://www.bvdinfo.com/Products/Company-Information/National/Bel-First. Accessed Jan. 10, 2013.

MODELLING AND SIMULATION OF ELECTRIC VEHICLES

M. GHEAMALINGA, R. PUHALSCHI, B. MIHALESCU, Al. GANZIUC
University POLITEHNICA of Bucharest
313, Splaiul Independentei, Sector 6
RO 060042 Bucharest, Romania
E-mail: mina.gheamalinga@icpe-actel.ro

C. VASILIU
RENAULT TECHNOLOGIE ROUMANIE
2/III Pipera Blv, North Gate Business Centre
RO 077190 Voluntari, Romania
E-mail: catalin.vasiliu.c@gmail.com

KEYWORDS
Modeling and simulation, energy systems, electric vehicle batteries management

ABSTRACT

The paper contains an assessment of the simulation models used in the early stage of an electrical vehicle design, and presents some experimental activities aiming to create a test bench for validating these models. Computational methods, control software, design problems, and experimental validation are shortly presented. From an industrial point of view, the main idea of the paper is the use of high-quality industrial electronic and electromechanical components only, in order to avoid manufacturing difficulties, high prices, and the need of the manufacturer permanent technical assistance in the preliminary phase of building a prototype for an electric vehicle with different kind of batteries. This target needed a new approach of the design and tuning complex nonlinear speed controllers by numerical simulation.

INTRODUCTION

While electric motors and the idea of using them to power a vehicle is almost two centuries old, only recently, due to the economic and environmental conditions, has the electrical car began to be considered as a viable alternative to the internal combustion engine powered cars, with multiple large automotive producers like Toyota introducing electrical models into series production (Pistoia 2010; Dragne 2010; Bățăuș 2011; Cioranu 2012).
In addition to this shift toward new technologies, the design methodology used in industry has also experienced a revolution, caused by the rise of simulation software. The traditional design workflow was built mainly along the following sequence: requirements> design> implementation >testing and validation (McDonald 2010). This approach was rather prone to various problems and delays, mainly due to the fact that often problems with the product would not be discovered until the testing phase.
The new approach, called Model-Based design (Lebrun 2009; LMS 2012) is based on developing models and performing simulations of the proposed product, system or solution, which allow engineers to validate and verify the specifications against the requirements, and study the system's behavior in order to spot eventual issues long before the actual practical implementation. This approach has proven both time as well as cost-effective.

With these considerations in mind, the authors have decided to illustrate a Model-Based approach toward the design of the drive system of an electrical vehicle (Fig. 1).

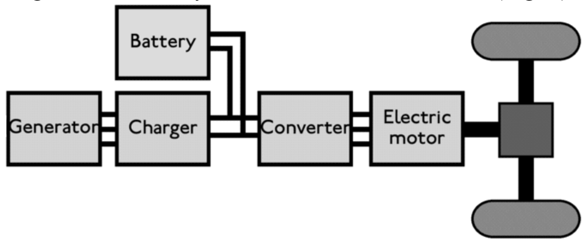


Fig. 1 Diagram of a generic electric vehicle drive system

Initially, the authors have analyzed two well-known models in the field, in order to provide a basis of experimental data against which the results obtained by two Hardware-in-the-Loop test stands designed in the research team and detailed later in the paper could be validated.

DRIVING CYCLES

Driving cycles are sets of data points representing the speed of a vehicle versus time. Some are developed theoretically, (as it is preferred in the European Union) while others are direct, measurements of driving patterns deemed representative.
Driving cycles are extremely useful in propulsion systems simulations (like the subject of this paper) as they allow measuring and predicting the performance of the drive system under conditions very similar to the real ones.
A typical example of a driving cycle (ECE-15 urban driving cycle) can be seen below in fig. 2.

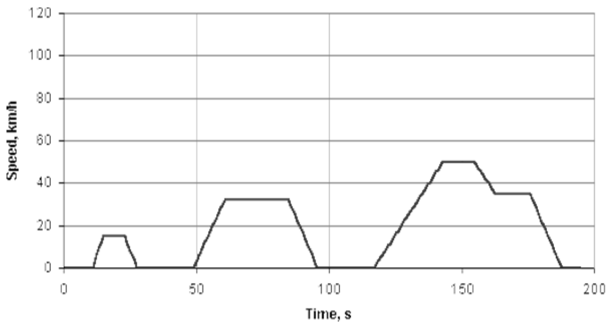


Fig. 2 ECE-15 urban driving cycle

In addition to the driving cycle, most abstract models, including the two models analyzed in this paper, also require a torque vs. time input, which is an abstract representation of the various resistive forces the drive system encounters (frictions, ground resistance, aerodynamic resistance, road profile etc.)

SIMULINK MODEL

The first model (Appendix A) is a SIMULINK model developed by David McDonald of Lake Superior State University and has been used to study the energy flow of the drive system under specific drive cycle and load profiles. The model consists of a simplified DC permanent magnet motor model; a motor control unit that regulates the voltage and current to meet the motor's needs; a simplified (voltage source with internal resistance only) battery model; a PI controller that regulates the entire process by acting upon the motor control unit's gain factor. One of the more representative simulations performed on this model uses the driving cycle and torque vs. time inputs presented below in fig. 3 and 4.

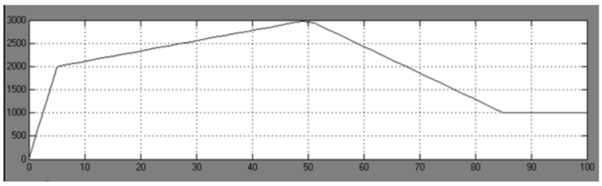


Fig. 3 Driving Cycle

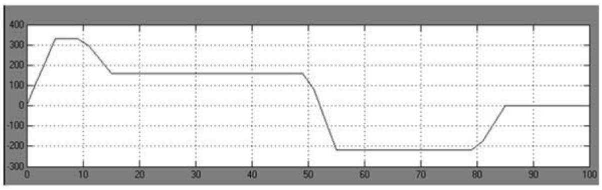


Fig. 4 Torque vs. time input

As it can be seen in fig. 5, the power output of the electric motor reaches both positive and negative values during the simulation. This illustrates the two phases of any modern electric drive system: motoring and regeneration.

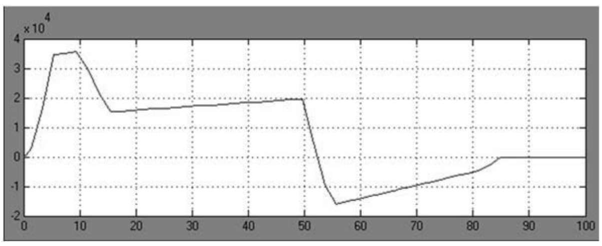


Fig. 5 Motor power

During the motoring phase (when the power is positive), the motor absorbs power from the electrical circuit, discharging the battery, and outputs mechanical work acting upon the load (moving the car).

During the regeneration phase (which usually occurs when the vehicle is braking), the motor absorbs mechanical energy from the load (which in a conventional vehicle would be lost in the braking system) and works as a generator, outputting electrical power and charging the battery. The battery has been considered to have a constant voltage of 200 V, while the output voltage depends on the power losses on the internal resistance. The PI controller however operates to make sure that the output voltage is maintained as close as possible to 200 V to prevent battery damage and life reduction) while still meeting motor power demands. The output voltage of the battery, as well as the error (in %) versus the required 200 V can be seen in fig. 6 and 7.

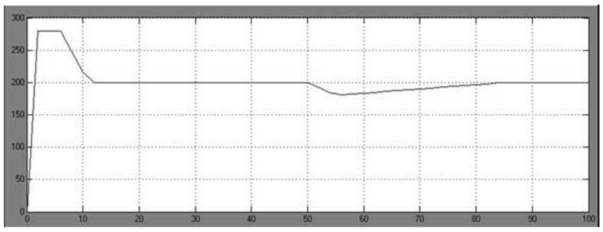


Fig. 6 Battery voltage

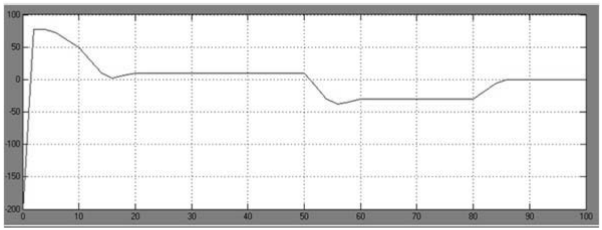


Fig. 7 Battery error

The voltage spike that can be seen at the beginning (topping at 76.9% over the desired 200 V) is due to both increased motor power demands during startup as well as numerical simulation inaccuracies (the values are very close to the start of the simulation, control loops might not have stabilized yet).

While this model is very useful in studying the energy flow in an electric motor drive system, without significant modifications it cannot provide detailed battery information like the state of charge (which prevents simulating a drive cycle that includes charging from the power grid or making any considerations regarding vehicle autonomy) or temperature (battery overheating during charging is a serious concern in regard to electric vehicles).

AMESIM MODEL

Rev 10 of AMESim (LMS 2012) includes several models for electric vehicle powertrains, ranging from simple ones, very similar to the SIMULINK model presented above to highly complex ones (Appendix B), including full vehicle model, multiple battery cells, auxiliary vehicle systems, and thermal modeling.

A representative simulation using this model addresses the two main concerns expressed above concerning the SIMULINK model: monitoring battery state of charge and temperature.

The driving cycle (fig.8) consists of standing still connected to the power grid for battery charge for 30 minutes, and then driving constantly at 25 m/s (90 km/h) for 30 minutes.

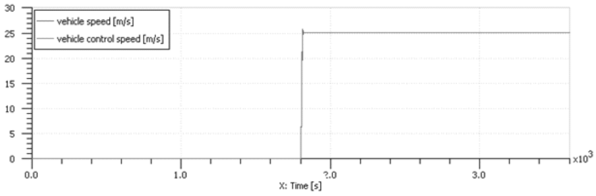


Fig. 8 Driving cycle

The torque vs. time input can be seen in fig. 9.

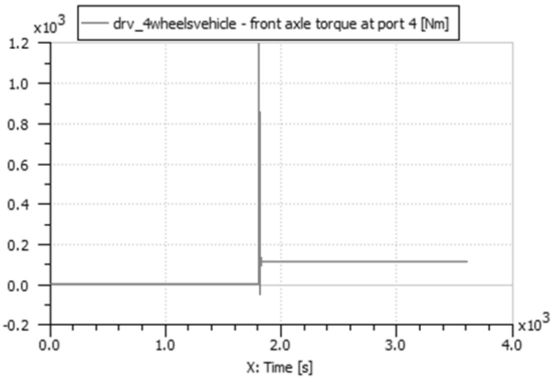


Fig. 9 Torque vs. time input

By monitoring the battery's state of charge (fig. 10) during both the charge and discharge phases, the vehicle's autonomy can be deduced (135 km, 90 min driving time).

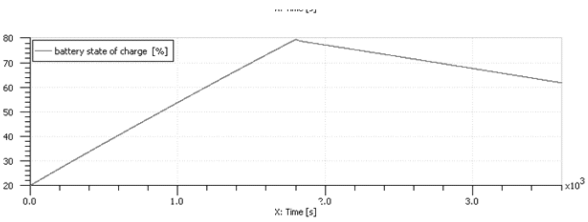


Fig. 10 Battery state of charge

In order to analyze the battery's thermal behavior, a very unfavorable case has been analyzed: idling during charge and driving in a hot climate, the ambient temperature in the battery's compartment has been set to 40 degrees Celsius. The battery is cooled both naturally via the air in the battery compartment as well as forced if necessary via the vehicle's air conditioning system (which also insures the cooling of the cabin). The evolution of the battery and cabin temperature is presented in fig. 11. At the start of the simulation, both temperatures are equal with the ambient temperature (40 degrees Celsius), as the vehicle is assumed to have been stopped without power for an indefinite period. In the first part of the driving cycle, because the vehicle is standing still while charging, the effectiveness of the battery cooling elements is small, because the air flow is insufficient (fig. 12).

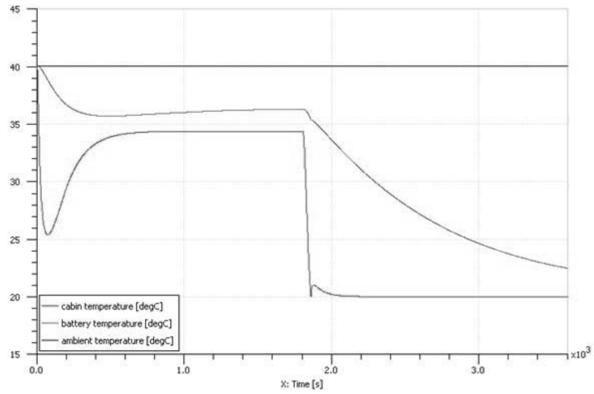


Fig. 11 Battery, cabin and ambient temperature

Therefore, part of the power of the air conditioning system is directed toward keeping the battery temperature within acceptable limits, which is noticeable (fig. 11) in the cabin temperature increasing from 25 degrees (achieved shortly after start) to over 35 degrees, value which remains constant until the vehicle starts moving. Once the vehicle is in motion, the battery cooling improves drastically, for two reasons: the airflow over the cooling elements almost doubles (fig. 12) and the heat dispersed by the battery during discharging is, unlike charging, negligible (fig. 13).

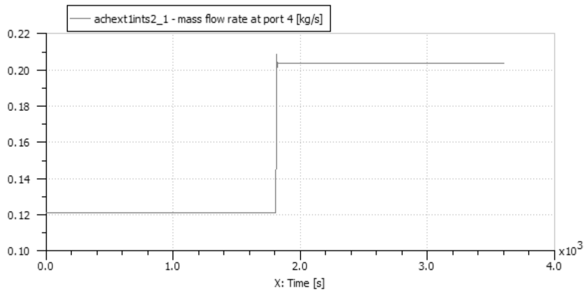


Fig. 12 Air flow through battery cooling elements

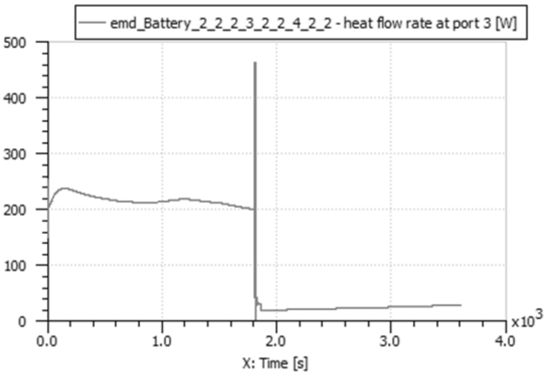


Fig. 13 Battery heat flow

As a result, battery temperature drops, but the air conditioning system is also 'freed' to cool down the cabin, where the temperature drops to the preset 20 degrees Celsius.

HIL TEST STAND

The simplified schematic of a complex HIL test bench (Vasiliu C. 2011) is presented below in Appendix C and fig. 14 (the stand itself).

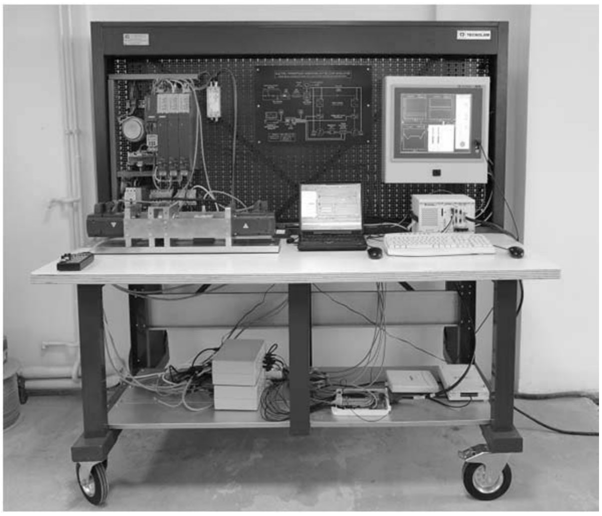


Fig. 14 Front view of the HIL test bench

The stand is built around a pair of brushless AC motors, connected together via a mechanical coupling. The coupling also includes a mass (disk) that simulates the inertia occurring in a real drive system. One motor is controlled with a speed reference, and it plays the role of the actual electric motor of the vehicle, providing power to the wheels. The other motor is controlled with a torque reference, and it simulates the load, offering mechanical resistance to the first motor.

The power is provided from a connection to the fixed power grid of the laboratory, through a common transformer and a separate inverter for each motor. The inverters are commanded via analog signals, as this has proven the only solution that can insure reliable control in real time at the speed required by this simulation.

Three National Instruments PXI computers provide the real time command and control processes. The first one acts in a manner similar to the ECU (engine control unit) of a real vehicle, acting upon the speed-controlled motor (the one simulating the actual electric vehicle motor). The second one contains the AMESIM model of the entire vehicle and acts upon the second electric motor, generating the load for the first engine (resistive torque) computed by the model depending on the actual simulated vehicle driving conditions. The third PXI computer acts as a data acquisition platform, handling the reading of all electrical parameters. All three machines are connected to a fourth, non-real-time computer whose role is to provide a graphical interface for the user for altering parameters, viewing and storing the measurements results. This is necessary because, while the PXI machines do have the hardware capability for video output, the NI real-time operating system does not support such facilities, and as such, any graphical interface needs to be implemented on a separate machine. The simulation setup and data acquisition interfaces can be seen in fig. 15.

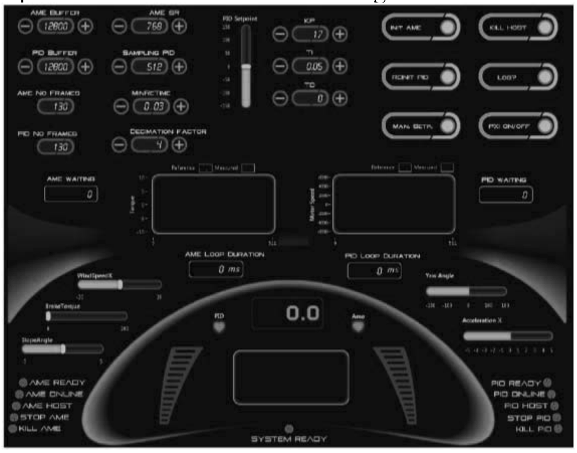


Fig. 15 Simulation setup interface

The test stand has allowed the authors the possibility to study the behavior of an electric vehicle drive system under various driving conditions (driving cycles), and some representative results are presented below.

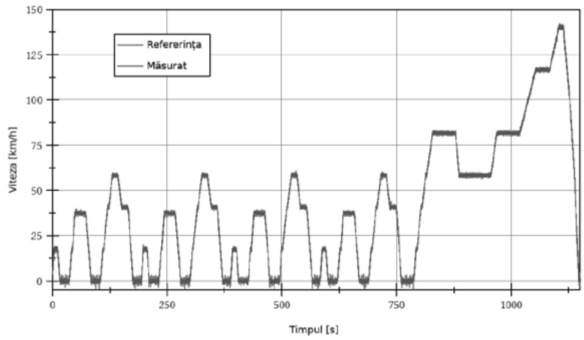


Fig. 16 New European Driving Cycle and system response

Fig. 16 presents the system's response (blue) during a typical driving cycle (red). As it can be seen, the response stays very close to the reference (max. 0.10 km/h difference), which is a proof of the system's performance.

The calculated torque (red) vs. the achieved torque (blue) in same conditions is presented in fig. 17.

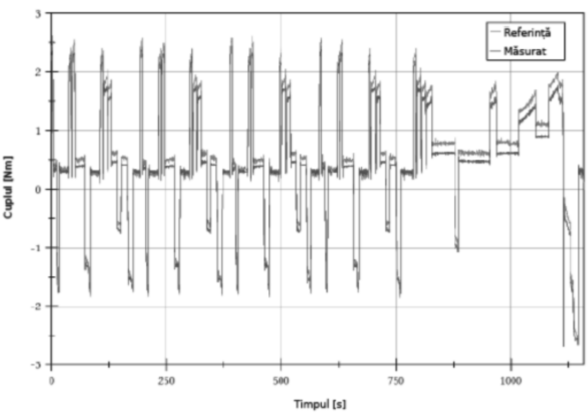


Fig. 17 Torque reference and response for NEDC

Fig. 18 and 19 present several other key variables that are monitored during the simulation.

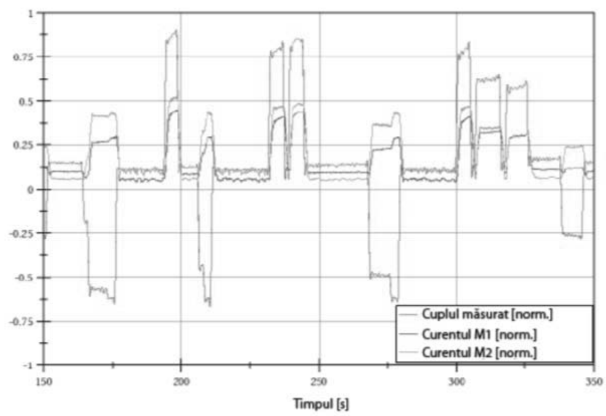


Fig. 18 Engine currents and torque evolutions

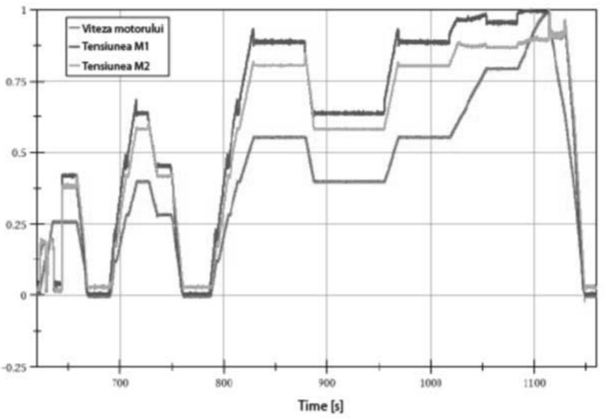


Fig. 19 Engine voltage and speed evolutions

As it can be seen, the values obtained through HIL simulation follow the established patterns that have been highlighted in the two simulation models described previously.

CONCLUSIONS

Starting from well-known and validated models and system representations, the authors have tried to identify the main characteristics as well as design challenges related to design of electrical vehicle drive systems. This knowledge has allowed the authors to develop a modular HIL test stand using high-quality industrial electronic and electromechanical components only, which allows reliable testing and analysis of various parameters of an electric vehicle drive, sizing of motors and batteries, tuning control loops etc. (Vasiliu 2011). The battery management optimization in the frame of an EV needs a special arrangement of the HIL system. Such a test bench (Annex D) was designed and built by the authors (Gheamalinga 2013). A cascade of the rectifiers and invertors is integrated with a mechanical arrangement including two brushless reversible electrical machines.

ACKNOWLEDGEMENTS

Authors are grateful to Professor Nicolae Vasiliu, member of the Romanian Technical Science Academy, for the modern engineering vision and full support in developing challenging researches in the field of HIL simulation for electrical vehicles.

REFERENCES

Băţăuş, M.V. 2011. "Hydraulic Control Systems for Automotive Mechanical Transmissions". Ph.D. Thesis, University POLITEHNICA of Bucharest.

Cioranu, C.N. 2012. "Researches on the Redundant Hydraulic Control Systems." Ph.D. Thesis, University POLITEHNICA of Bucharest.

Dragne, F.D. 2010. "Researches on numerical Modeling and Simulation of Automotive Hydraulic Systems with Application on an ABS/ESP Braking System." Ph.D. Thesis, University POLITEHNICA of Bucharest.

Gheamalinga, M. 2013. ."Battery management of the Electric Powertrains". Ph.D. Thesis, University POLITEHNICA of Bucharest.

Lebrun, M., Vasiliu D., Vasiliu, N. 2009. "Numerical simulation of the Fluid Control Systems by AMESim". Studies in Informatics and Control with Emphasis on Useful Applications of Advanced Technology, Volume 18, Issue 2, p.111-118.

McDonald D. 2012. Electric Vehicle Drive Simulation with MATLAB/Simulink, Proceedings of the 2012 North - Central Section Conference.

McDonald D. 2011. AEV 5010 Introduction to Advanced Electric Vehicles. Lake Superior State University at Sault Sainte Marie.

Mitroi, M.Al. 2013. "Researches on the Modeling and Real Time Simulation of the Electrohydraulic Control Systems." Ph.D. Thesis, University POLITEHNICA of Bucharest.

Muraru, V., Vasiliu, N., Vasiliu, D. 2000. "Structural Analysis of the Proportional Solenoids for Control Systems using ANSYS Software". World ANSYS Conference, Pittsburgh, U.S.A.

Pistoia G. 2010. "Electric and hybrid Vehicles", Elsevier.

Vasiliu D., Vasiliu G.C., Vasiliu N. 2012. "Using Automation Studio for the Design of the Fluid Control Systems". POLITEHNICA PRESS, ISBN 978-606-515-397-4.

Vasiliu N., Vasiliu G.C.,Vasiliu D., Irimia P.C. 2012. "Modern Developments of the Hydraulic Transmissions in Automotive Technology". ROMANIAN ACADEMY OF TECHNICAL SCIENCES and AGIR PRESS HOUSE, ISSN 2066-6586.

Vasiliu C. 2011. "Real Time Simulation of the Electric Powertrains". Ph.D. Thesis, University POLITEHNICA of Bucharest.

***Imagine, 2001. "Numerical Challanges posed by Modeling Hydraulic Systems". Technical Bulletin 114.

***LMS, 2012. "AMESIM - Advanced Modeling and Simulation Environment", Release 12, Leuven.

WEB REFERENCES

http://www.boschrexroth.com
http://www.famictech.com/
http://www.lmsintl.com/
http://www.mathworks.com/products/simulink/
http://www.nhancetech.com/
http://www.dspace.com/

BIOGRAPHIES

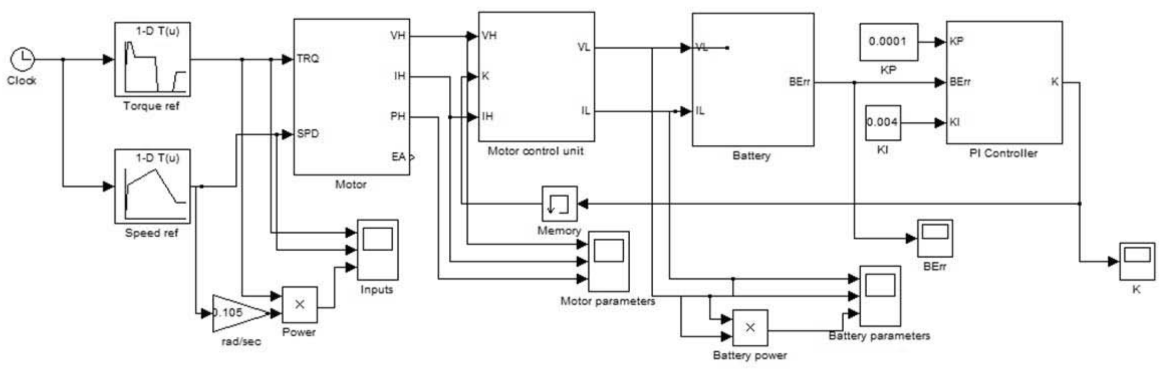
GHEAMALINGA MINA graduated in material science in 2004, and prepared a M.Sc. at I.N.P. Toulouse in 2006. He elaborated a Ph.D. thesis in the field of the electric vehicle batteries management in the frame of the Power Engineering Faculty of the University POLITEHNICA of Bucharest. He is currently a scientific researcher in the Department of Electric Drives and Automation Engineering from ICPE ACTEL Company, with works in the field of modeling, simulation, and experimental identification of the electromechanical control systems.

PUHALSCHI RADU graduated in Applied Computer Science from University POLITEHNICA of Bucharest in 2009, and a MD in Advanced Hydraulic and Pneumatic Systems in 2011. After a stage of web designer at HP Germany, he began a PhD research work on Real-Time Simulation of hydraulic systems at the Fluid Power Laboratory from the University POLITEHNICA of Bucharest.

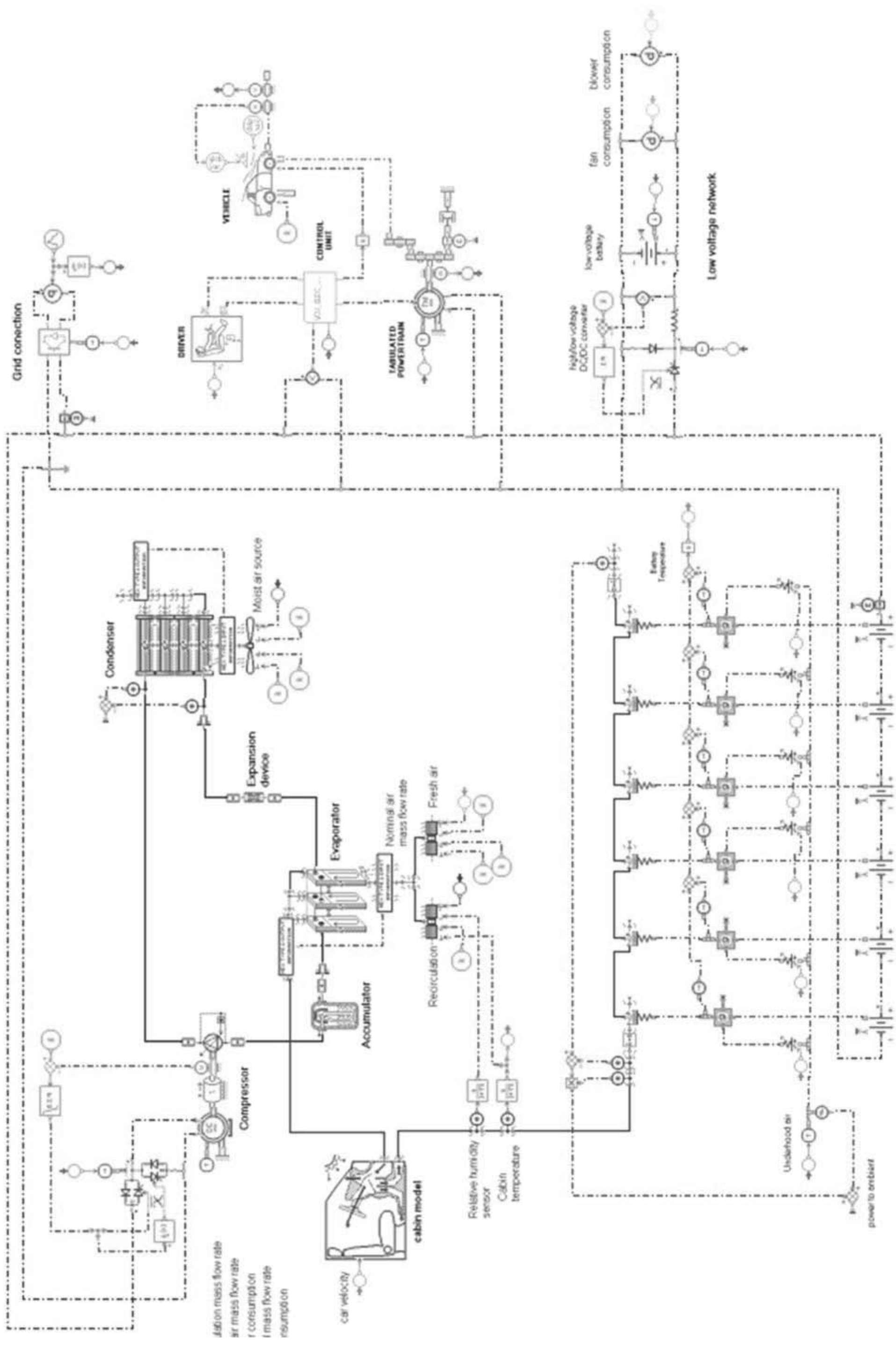
BOGDAN MIHALESCU graduated in Hydropower Engineering from University POLITECHNICA of Bucharest in 2007. He became assistant professor and PhD student in the Fluid Power Laboratory from the same university. His main areas of interest include modeling, simulation and dynamic identification of hydraulic systems. He is a certified associated developer of National Instruments Corporation.

GANZIUC ALEXANDRU graduated in electronic engineering in 2010, and received the M.Sc. degree in Advanced Hydraulic and Pneumatic Systems in 2011 in the frame of the Fluid Power Laboratory from the Power Engineering Department of the University POLITEHNICA of Bucharest. He developed a carrer in R&D, having a deep education in digital control.

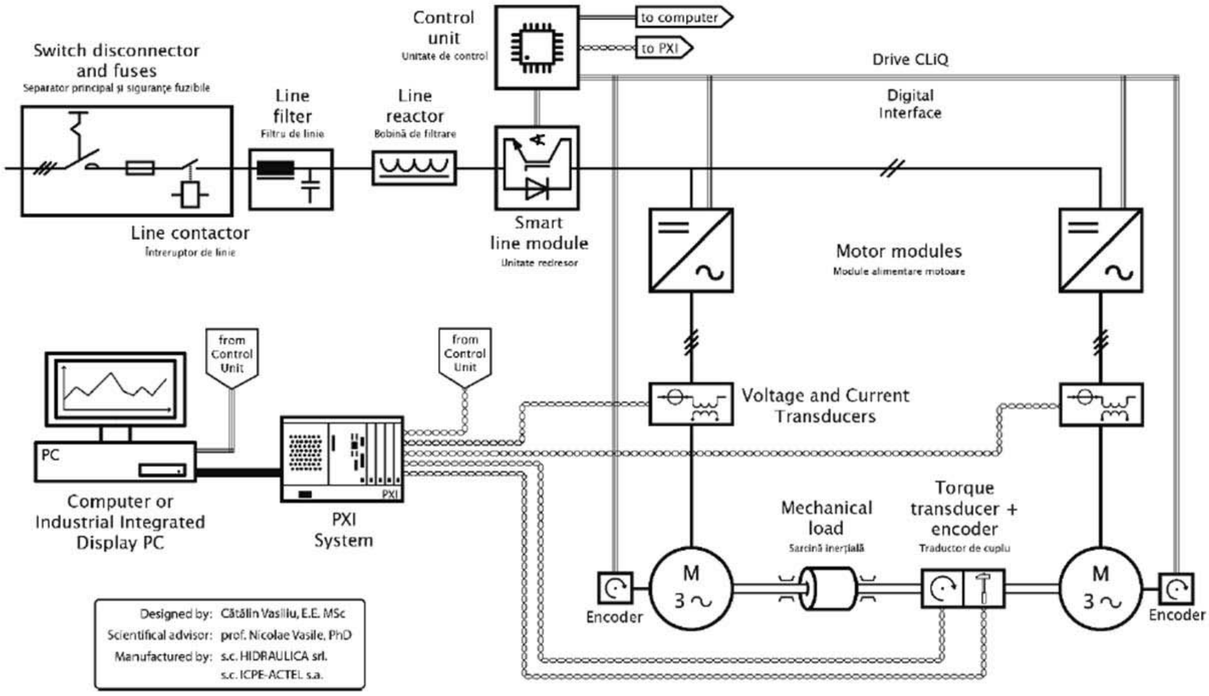
CĂTĂLIN VASILIU graduated in electrical engineering in 2008 in the frame of the Power Engineering Faculty of the University POLITEHNICA of Bucharest, and prepared a M.Sc. at the same university. He elaborated a Ph.D. thesis with LMS International in the field of the electric vehicle HIL simulation. He is currently a development engineer in Renault Technologie Roumanie.



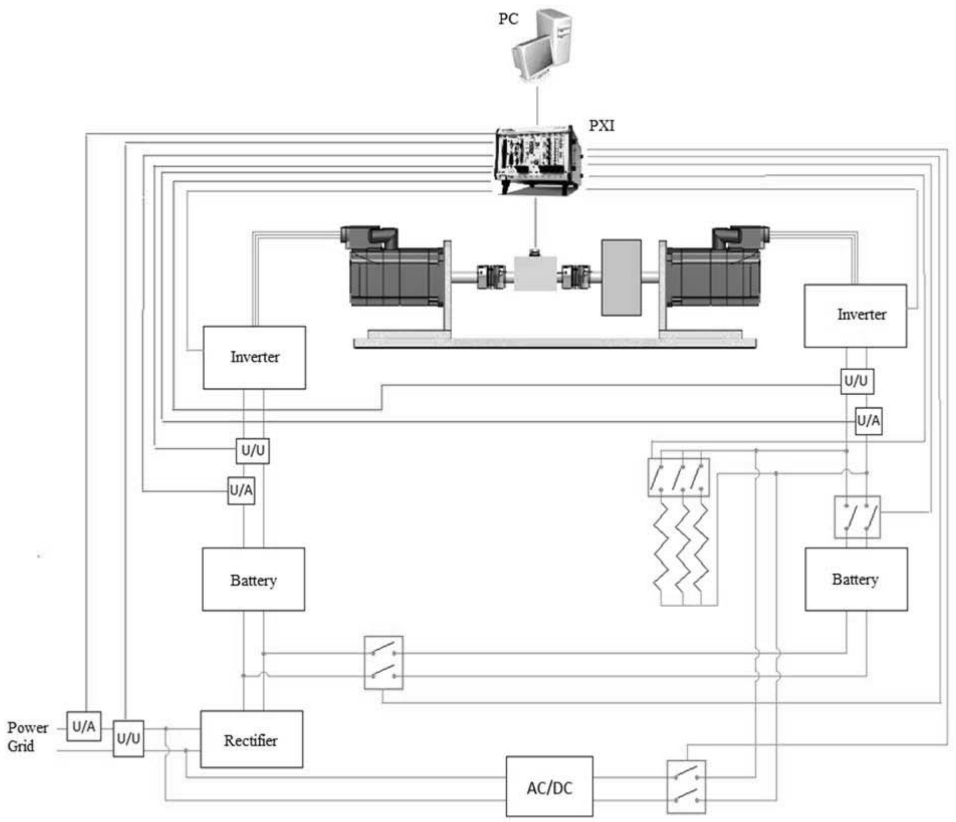
APPENDIX A. Basic Simulink simulation network for an EV



APPENDIX B. AMESIM network for a complete model of an EV



APPENDIX C. Test bench diagram for Hardware-in-the Loop simulation of an electric powertrain (Vasiliu 2011)



APPENDIX D. Test bench diagram for battery management of an electric powertrain (Geamalinga 2013)

SIMULATION AND TUNING A DIGITAL SERVOMECHANISM

Alexandru GANZIUC, Mina GHEAMALINGA, Szilard FEHER, Radu PUHALSCHI, Daniela VASILIU
University POLITEHNICA of Bucharest
313 Splaiul Independentei, Sector 6
RO 060042 Bucharest, Romania
E-mail: vasiliu@fluid-power.pub.ro

KEYWORDS

Simulation, digital servomechanism, fine tuning, electromagnetic compliance

ABSTRACT

The paper contains an assessment of the static, dynamic, and electromagnetic compliance of the industrial high-response valves with integrated axis controller with CANopen facilities. Two series of experiments were performed in order to find out the difference between two ways of generating a nonlinear steady state characteristic: shaping the metering windows of the power stage (for analog servovalves) and changing the slope of the curve around the null region by the aid of a function implemented in a EPROM memory of a digital servovalve. Another important objective of the paper is the study of the impact of the eddy electromagnetic fields on both types of devices (analog and digital ones). Digital devices with special connections are well protected against any source of electromagnetic disturbances.

INTRODUCTION

The possibility of connecting many electrohydraulic axles by a single bus in the frame of a complex control system is an old "dream" of the mechatronic designers. Since 1990 some innovative companies like Moog (Boes 2003), developed flapper-nozzle two stages servovalves and connected by wires or optical fibers. In the last decade, the same companies integrated the bus controller in DDV's: (Rexroth 2005) and (Parker 2004).

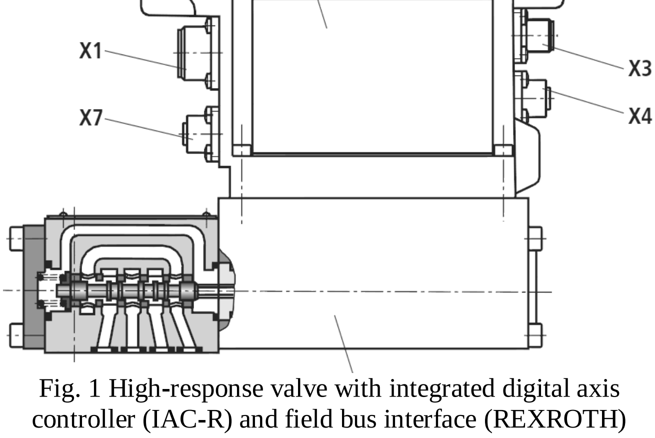


Fig. 1 High-response valve with integrated digital axis controller (IAC-R) and field bus interface (REXROTH)

The REXROTH IAC-R valve (Integrated Axis Controller on the basis of high-response valves) is one of the last generation of digital high-response valve with integrated axis controller with the following functionalities: flow control; position control pressure control; p/Q function; function; optional position/pressure and position / force control; NC functionality; the command value can alternatively be provided via an analog interface (X1) or via the field bus interface (X3); the actual value signals are provided via an analog interface (X1) and can additionally be read out via the field bus (X3); the controller parameters are set via the field bus; separate supply voltage for bus/controller and power part (output stage) for safety reasons PC program WinHPT.

To implement the project-planning task and to parameterize the IAC-R valves, the user may use the commissioning software WinHPT. This allows diagnosis, programming of NC functionality, parameterization; comfortable data management on a PC, the use of a PC operating systems like Windows 2000 or Windows XP.

The following additional functions are available: ramp generator; internal command value profile; release function analog/digital; error output 24 V (e.g. as switching signal to PLC/logic and further valves); control output adjustment; dead band compensation; zero point correction; valve inflection compensation; friction compensation; direction-dependent gain.

The digital integrated control electronics enables the following fault detection: cable break sensors; under voltage; temperature of the integrated electronics; communication errors; watchdog.

The servovalve steady state behavior can be set up in a digital manner according the controlled process needs (fig. 2) or in a "hardware manner" (fig. 3) used mainly in two stages flow valves.

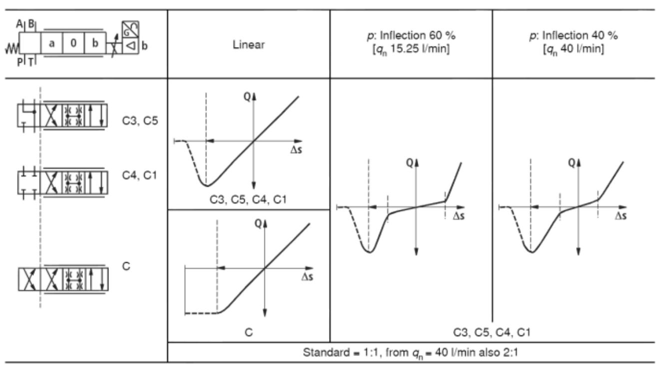


Fig. 2 Different types of characteristics of IAC-R

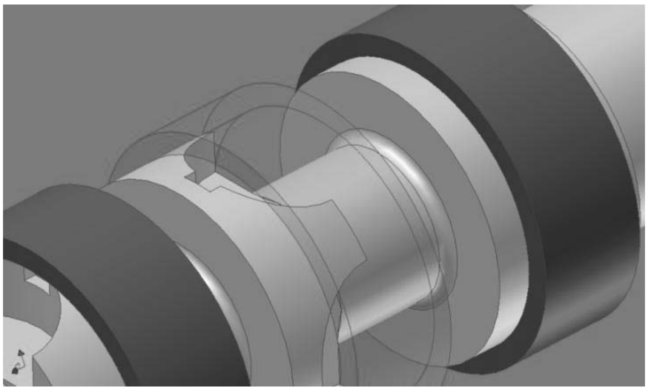


Fig. 3 Spool with shaped metering windows for two slope steady state characteristics (Vasiliu 2009)

This paper presents the digital servo system experimental identification and the comparison with the analog model. The influence of a typical electromagnetic disturbance generator composed by an asynchronous motor driven by a frequency converter is also studied.

REFERENCE ANALOG MODEL

The reference model of an analog electrohydraulic servo system (Daniela Vasiliu 1997) has been studied by a SIMULINK program (Appendix A) which includes the following equations.

a) The steady-state characteristics of the servovalve (four ways, critical centre, spool valve):

$$Q_{SV}(x,p) = c_d A(x)\sqrt{\frac{p_S - P}{\rho}} \quad (1)$$

Here x is the spool displacement from the neutral position; P - pressure difference between the ports of the hydraulic cylinder; A(x) – metering ports surface; c_d – discharge coefficient of the metering ports; p_S - supply pressure (constant). The above relation can be written in the form

$$Q_{SV}(x,P) = K_{Qx} x\sqrt{1 - P/p_S} \quad (2)$$

where

$$K_{Qx} = c_d \pi d_s \sqrt{p_S/\rho} \quad (3)$$

is the "flow valve gain".

b) The spool motion equation.

The servovalves manufacturers specify for each device the transfer functions adequate to slow, normal and high-speed control process. For slow control process, the servovalve can be regarded as a proportional device, with a single constant - the displacement-current (voltage) gain:

$$K_{xi} = \left.\frac{\partial x}{\partial i}\right|_{x=0} \quad (4)$$

Spool follows the input current, i without any practical lag:

$$x = K_{xi} i \quad (5)$$

For normal control process, a servovalve can be regarded as a first order lag device:

$$\frac{x(s)}{i(s)} = \frac{K_{xi}}{T_{SV}s + 1} \quad (6)$$

The corresponding differential equation is:

$$T_{SV}\frac{dx}{dt} + x = K_{xi} i(t) \quad (7)$$

Here T_{SV} is the servovalve time constant. In high speed control process, a servovalve has to be described by a second or even third order lag device:

$$\frac{x(s)}{i(s)} = \frac{K_{xi}}{(s/\omega_n)^2 + 2s\zeta/\omega_n + 1} \quad (8)$$

where ω_n is the natural frequency, and ζ - damping coefficient.

c) The position transducer equation.

The modern inductive position transducers together with their bridges behave as first order lag devices; they have a very small time constant, which can be neglected for industrial electrohydraulic control process:

$$U_T = K_T y \quad (9)$$

where K_T is the transducer constant, and *y* – piston displacement from null position.

d) The error amplifier equation.

This stage computes the following error, ε as a difference between the input signal, U_i and the position transducer output, U_T, and applies the PID control algorithm to find the solenoid control voltage, U_c:

$$U_c(s) = \varepsilon(s)K_P[1+1/(sT_i) + sT_d/(\tau s + 1)] \quad (10)$$

The latest control algorithms are based on the fuzzy and adaptive fuzzy systems theory [7].

e) The servocontroller current generator equation.

The current generator of the servocontroller is so fast than it can be regarded as a proportional device:

$$i = K_i U_C \quad (11)$$

where K_i *[A/V]* is the "medium" conversion factor.

f) The continuity equation.

This equation represents the connection between the servovalve flow and the derivative of the pressure drop across the hydraulic cylinder:

$$Q_{SV} = A_p\dot{y} + K_l P + \frac{A_p^2}{R_h}\dot{P} \quad (12)$$

where A_p is the piston area; K_l - leakage coefficient between the motor chambers; R_h - hydraulic stiffness of the motor,

$$R_h = 2\frac{\varepsilon_e}{\mathbf{V}_t}A_p^2 \quad (13)$$

Here ε_e is the equivalent bulk modulus of the oil and $\mathbf{V}_t$ is the total volume of the oil from the motor and the hydraulic connections.

g) The piston motion equation.

The pressure force F_p have to cover the load, modelled by a preloaded spring force F_e, inertia of the moving parts, m_e and the friction force, F_f:

$$m_e\ddot{y} = F_p - F_e - F_f \quad (14)$$

where

$$F_p = A_p P \quad (15)$$

$$F_e = 2K_e(y + y_{0e}) \quad (16)$$

The friction force has mainly a static component, F_{fs} and a viscous one, F_{fv}:

$$F_{fs} = F_{fs0}\,sign\,\dot{y} \quad (17)$$

$$F_{fv} = K_{fv}\dot{y} \quad (18)$$

A typical response of such a system for a small input is shown in fig. 4. It was obtained with a P controller only.

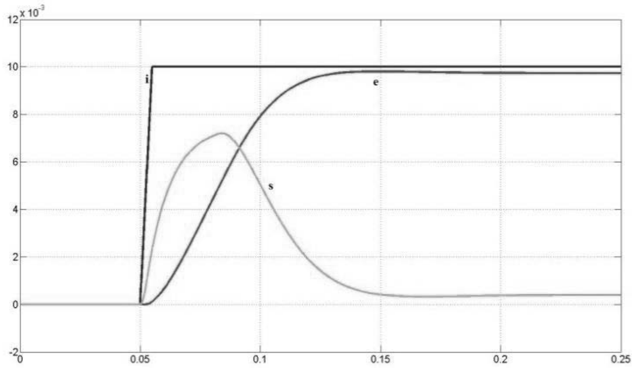


Fig. 4 Response of a typical analog servosystem for U_i=1.0V (y_i=10mm), with a slope of 50ms, K_{Qx}=0.66 m^2/s, T_{SV}=0.01ms and K_p= 3 (x is reaching 50% from the maximum value)

EXPERIMENTAL RESEARCH

The authors studied the behavior of a IAC-R servovalve in the frame of a test bench with the same structure as the analog model described above. The hydraulic diagram is presented in fig. 5, and a partial view - in ANNEX 2 (Ganziuc 2013). Three series of experiments were performed in close loop with a spring-mass load, typical for the eleron of a small size aircraft: a) step input response with 10% analog input via the interface X1; the input signal was provided by a STANFORD universal signal generator, numerically controlled (fig. 6); b) step input response with 10% analog input via the the field bus interface (X3); the input signal was provided by the software WinHPT; c) sine input response (fig. 8) with 10% analog input via the the analog interface X1; the input signal was provided by a STANFORD universal signal generator, numerically controlled and the supply pressure was set at 100bar.

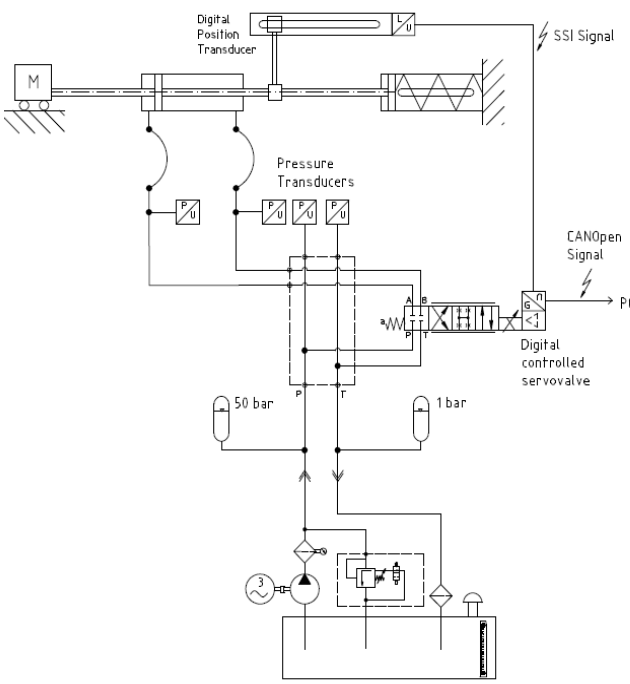


Fig. 5 Test bench scheme for IAC-R digital servovalve

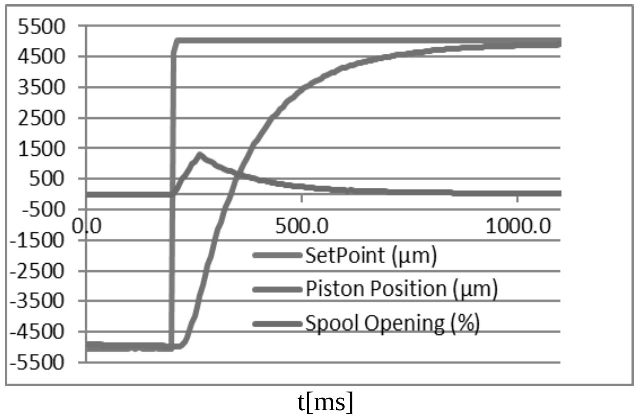


Fig. 6 Response of the servosystem to an analog step input U_i=1.0V (y_i=10mm), with a slope of 50ms, K_{Qx}=0.66 m^2/s and K_p= 3 (x is reaching 30% from the maximum value - 500 µm)

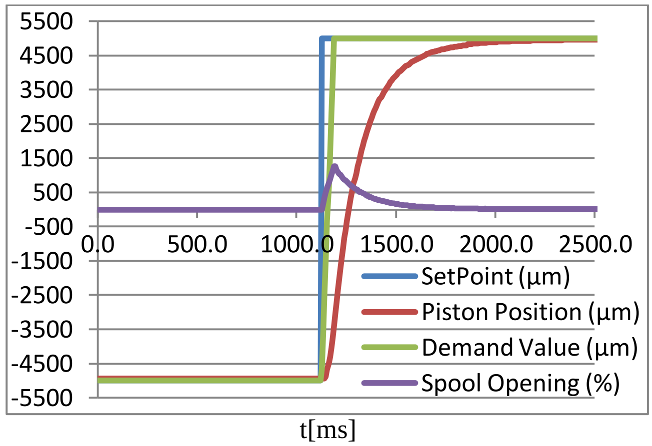


Fig. 7 Response of the servosystem to an digital step input U_i=1.0V (y_i=10mm), with a slope of 50ms, K_{Qx}=0.66 m^2/s and K_p= 3 (x is reaching 30% from the maximum value - 500 µm)

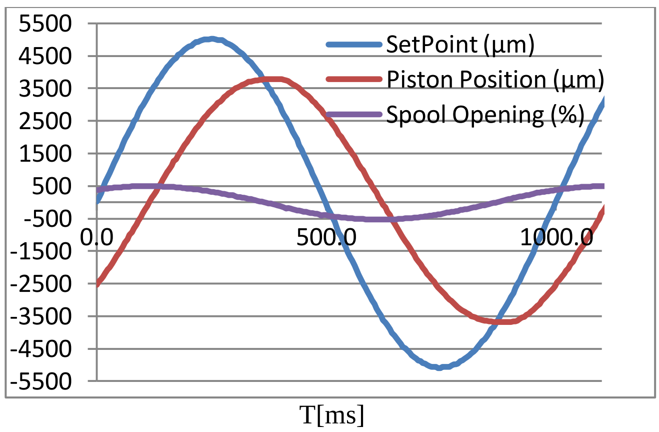


Fig. 8 Response of the servosystem to an analog sine input U_{imax}=1.0V (y_{imax}=10mm), K_{Qx}=0.66 m^2/s and K_p= 3 (x is reaching 10% from the maximum value - 500 µm)

The above diagrams show a better dynamics of the digital servovalve for digital inputs. The analog input introduces a delay because of the analog to digital conversion needed for the microcontroller operation. The APPENDIX C and D show a good overall accuracy over the entire piston stroke. An error of 61µm for a piston travel of about 100 mm is small enough for any industrial application. The frequency response seems to be slow (fig. 8), because the time needed for interpolation the path for any 20 microns (the digital position transducer resolution) is much greater than the one needed for performing a step input. The attempts to disturb the servovalve operation by an eddy electromagnetic field produced by an asynchronous motor driven by an ABB frequency converter failed. At the same time, a pure analog OBE servovalve with a voltage control shows a clear sensitivity for a common electromagnetic field generated by a strong industrial mobile communication device. This problem can easily be solved by replacing the voltage input by a current one (Muraru et al. 2000).

CONCLUSIONS

A long series of experiments led the research team to the following conclusions: the servovalve with integrated digital axis controller and field bus interface IAC-R concentrate all the functions of a servo controller and an analog servovalve in a very compact component with many setup options; the static performance practically depends on the resolution of the digital feedback position transducer only; the dynamics of the servovalve covers all the industrial applications requirements for step and linear inputs; the follow ability of a digitally defined path still remains to clarify by extensive experimental researches; the high-speed response of the digital servovalves creates the possibility to implement them in many applications with a small amount of programming activities. One of the most important application domain seems to be the dynamic simulators with Hardware-In-the-Loop, widely used in aerospace and automotive developments (Köhler 2011, Vasiliu C. 2011, Mitroi 2013).
The possibility to define the steady state characteristic in order to obtain the best match with the process (fig. 9) allows the manufacturers to decrease the manufacturing costs by promoting the same design of the valve sleeve.

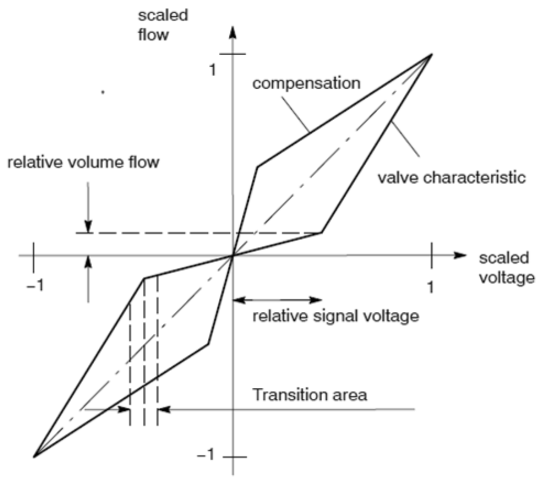


Fig. 9 Different software generated valve characteristics

ACKNOWLEDGEMENTS

Authors are grateful to Professor Nicolae Vasiliu, a member of the Romanian Technical Science Academy, for the modern engineering vision and full support in developing challenging researches in the field of digital control systems.

REFERENCES

***REXROTH. 2005. IAC-R valve (with CANopen). Operating manual RE 29090-B-01/06.05.

***REXROTH. 2010. High-response valve with integrated 1/22 digital axis controller (IAC-R) and field bus interface. Operating manual RE 29191/09.10.

***MOOG. 2005. Electrohydraulic Valves. A technical look.

***MOOG. Direct-Operated Servo Valves with Fieldbus Interface D636 and D637 Series ISO 4401 SIZE 03 AND 05.

***PARKER. 2004. Proportional Directional Control Valves, Bulletin Hy14-2599-M1/Us, Installation Guide.

Boes, C., Lenz W., Müller J., 2003. "Digital servo valves with fieldbus interface in closed loop applications", The Eighth Scandinavian International Conference on Fluid Power, SICFP’03, May 7-9, 2003, Tampere, Finland.

Ganziuc, Al. 2013. "Researches on the electromagnetic compatibility of the electrohydraulic control systems.". Ph.D. Thesis, University POLITEHNICA of Bucharest.

Köhler, T. 2011. Safer Landings Start With Smarter Servo Valves, Moog.

Mitroi, M.Al. 2013. "Researches on the Modeling and Real Time Simulation of the Electrohydraulic Control Systems." Ph.D. Thesis, University POLITEHNICA of Bucharest.

Muraru, V., Vasiliu, N., Vasiliu, D. 2000. "Structural Analysis of the Proportional Solenoids for Control Systems using ANSYS Software". World ANSYS Conference, Pittsburgh, U.S.A.

Vasiliu C. 2011. "Real Time Simulation of the Electric Powertrains". Ph.D. Thesis, University POLITEHNICA of Bucharest.

Vasiliu D. 1997. Researches on the Servopumps and Servomotors of the hydrostatic Transmissions. Ph.D. Thesis, University POLITEHNICA of Bucharest.

Vasiliu D., Vasiliu G.C., Vasiliu N. 2012. "Using Automation Studio for the Design of the Fluid Control Systems". POLITEHNICA PRESS, ISBN 978-606-515-397-4.

BIOGRAPHIES

Alexandru GANZIUC graduated in electronic engineering in 2010, and received the M.Sc. degree in Advanced Hydraulic and Pneumatic Systems in 2011 in the frame of the Fluid Power Laboratory from the Power Engineering Department of the University POLITEHNICA of Bucharest. He developed a carrer in R&D, having a deep education in digital control.

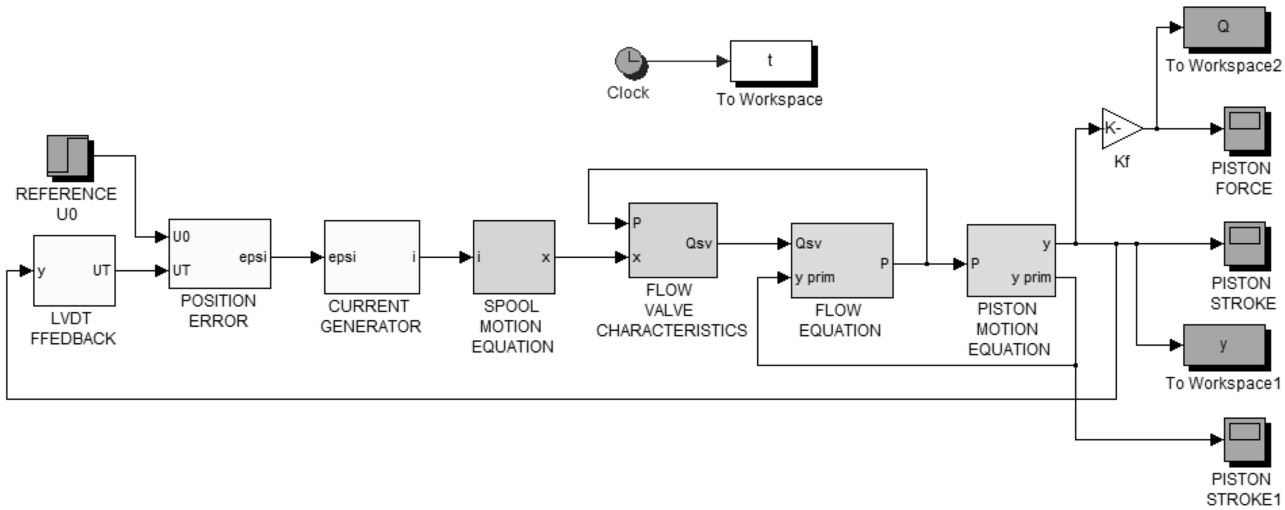
Mina GHEAMALINGA graduated in material science in 2004, and prepared a M.Sc. at I.N.P. Toulouse in 2006. He elaborated a Ph.D. thesis in the field of the electric vehicle batteries management in the frame of the Power Engineering Faculty of the University POLITEHNICA of Bucharest. He is currently a scientific researcher in the Department of Electric Drives and Automation Engineering from ICPE ACTEL Company, with works in the field of the design of the electromechanical control systems.

Radu PUHALSCHI graduated in Applied Computer Science from University POLITEHNICA of Bucharest in 2009, and a MD in Advanced Hydraulic and Pneumatic Systems in 2011. After a stage of web designer at HP Germany, he began a PhD research work on Real-Time Simulation of hydraulic systems at the Fluid Power Laboratory from the Power Engineering Department of the University POLITEHNICA of Bucharest.

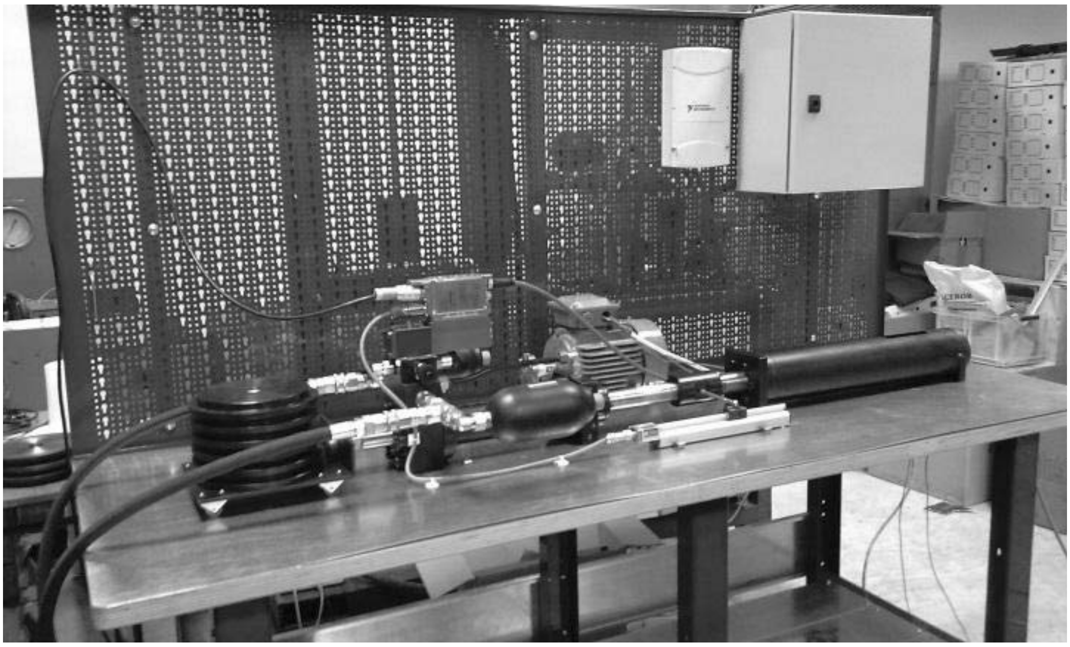
Szilard FEHER graduated in Mechanical Engineering from University POLITEHNICA of Bucharest in 2009, and a MD in Advanced Hydraulic and Pneumatic Systems in 2011. The same year he began a PhD research work on mobile electro hydraulic servo systems at the Fluid Power Laboratory from the Power Engineering Department of the University POLITEHNICA of Bucharest. He serves currently as technical advisor for the HIDRAULICA SRL from Brasov, a PARKER industrial applications developer.

Daniela VASILIU graduated in mechanical engineering in 1981 and prepared the Ph.D. thesis in the field of the dynamics of the hydrostatic transmissions. She is currently professor in the Department of Hydraulics, Hydraulic Machines and Environmental Engineering, head of Fluid Power Laboratory of the University POLITEHNICA of Bucharest. She works in the field of modeling, simulation, and experimental identification of the electro hydraulic control systems.

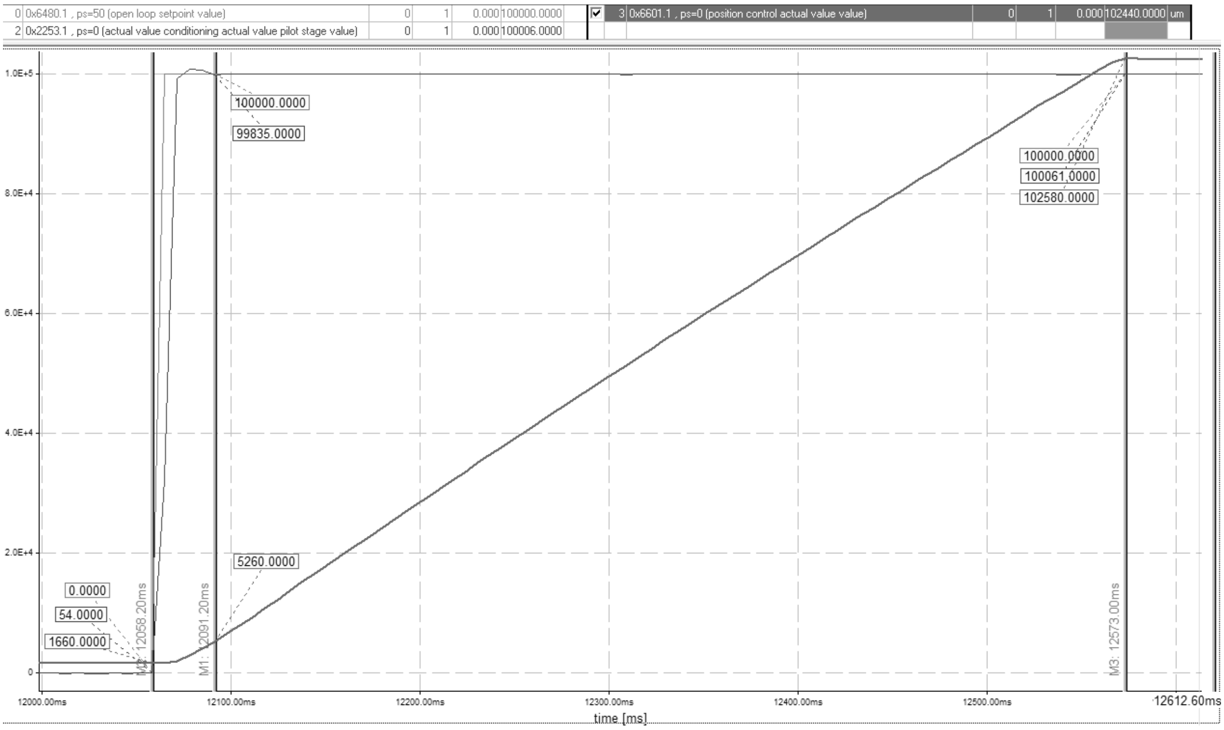
APPENDIX A. Basic Simulink simulation network for an electrohydraulic servomechanism (Vasiliu D. 1997, 2011)



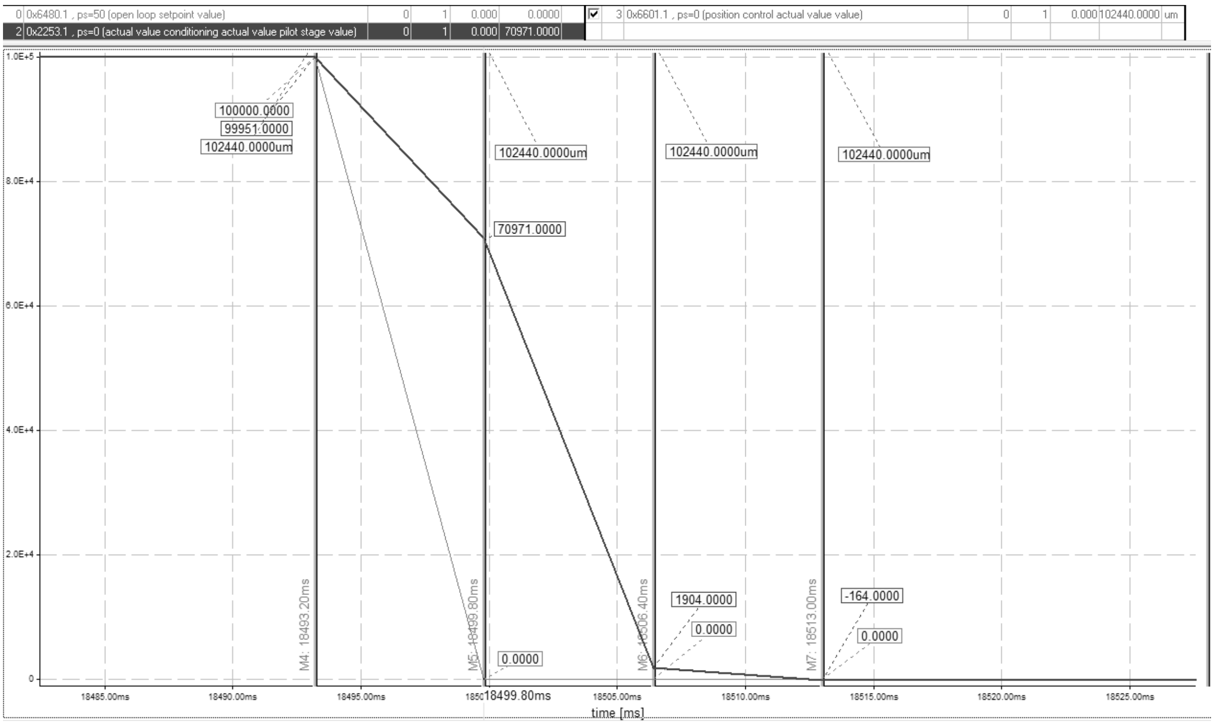
APPENDIX B. Test bench for digital servovalves in position control loop (Ganziuc 2013)



APPENDIX C. Servosystem response for a full stoke positive digital input with a ramp of 50ms.



APPENDIX C. Servosystem response for a full stoke negative digital input with a negative ramp of 50ms.



MODELLING, SIMULATION AND OPTIMIZATION OF HYDROPOWER PLANTS CASCADE BY GENETIC ALGORITHMS

Vlad Florin PÎRĂIANU, Nicolae VASILIU
University POLITEHNICA of Bucharest
313 Splaiul Independentei, Sector 6
RO 060042 Bucharest, Romania
E-mail: vlad.piraianu@gmail.com; vasiliu@fluid-power.pub.ro

Gheorghe IANA
SC Hidroelectrica SA, Rm. Valcea Branch
11, Decebal Street
RO 240265 Râmnicu Vâlcea, Romania
E-mail: gheorghe.iana@hidroelectrica.ro

KEYWORDS
Modeling and Simulation by Genetic Algorithms, Energy Market, Hydropower Plants, Income Optimization

ABSTRACT

Hydroelectric power plants (HPP) are playing a major role in any modern (competitive) electrical energy system. The hydropower systems have certain advantages compared to other energy production systems: they have the ability to accumulate hydraulic energy in reservoirs for different periods of times; they have the fastest response time in case of energy demand; they provide reliable and quality grid services (frequency regulation, voltage regulation an reactive energy balancing); the produced energy is competitive on any energy market based on financial targets. This paper presents a new method to improve the contribution of a cascade of power plants on a spot market. The above result was obtained by the aid of an optimization algorithm having as main objective to maximize the generated income by planning the energy production in high demand hours. To prove the ability of this original algorithm, a comparison for two operation scenarios is presented, showing the influence of the market prices. The comparison is based on the same hydrological regime, the same availability of the power plant cascade, and offers two results for different optimization objectives: maximizing the energy production or maximizing the generated income. The two objectives are significantly different: maximizing the energy produced requires keeping the level in the lakes at the highest availability in order to have maximum head and therefore working as much as possible at rated power of turbines, where maximizing the generated income requires to accumulate water during trading intervals where the energy is cheaper based on market data and generate maximum energy during trading intervals with high prices.

INTRODUCTION

The use of optimization, as a general concept, is becoming a top priority in every sector of a modern economy, and its use is more and more demanding when it comes to rationalization and efficient use of resources. Water is one of the most important resource available and how it is used is of high importance for any modern society to assure its needs. Therefore, the importance of optimal use of water has been increasing for the last decades together with the development of modern optimization techniques. In the devoted literature we find extensive research and evaluation of different methods of optimization specialized in reservoir system operations (Yeh 1985, Wurbs 1993, Chau and Albermani 2003, Lebadie 2004). Modern techniques have been used to develop reservoir optimal operation in the past years using different methods: dynamic programming (DP), linear programming (LP), nonlinear programming (NLP), fuzzy logic (FL), genetic algorithms (GA) etc. Determining the optimal strategy in order to operate a large number of reservoirs with multiple usage is a challenging task, mostly because of the complexity and nonlinearity of the system itself. Therefore the focus of the above researches was to acquire the solution of an optimized model (Banos et al. 2001). Two typical types of algorithms have been developed in the past years, which are the mathematic programming approach and the artificial intelligence optimization (AIO) approach. From the AIO approach, the Genetic Algorithms (GA) has been widely used in reservoir optimization (Holland et al. 2012, Chang et al. 2010, Chiu et al. 2007, Ferreira et al. 2012, Mallipeddi et al. 2010). They have enhanced the main algorithm with specific methods of performance increasing by constrains (penalty functions being the most used) and use in addition with other optimization methods and techniques (Lin et al. 2004, Nanakorn et al. 2001, Deb 2000, Runarsson and Meesomklin 2001, Chootinan and Chen 2006, Cheng et al. 2008, Xu et al. 2012). This paper presents the results obtained with a new optimization method based on GA combined with penalty function and restrictions in behavior of the level of the lakes with specific variation during the day.

ELECTRICITY MARKET IN ROMANIA

In Romania the electricity market is continuously changing encouraging the development of a competitive environmental for producers and suppliers. The Romanian electricity market operator (Opcom) coordinates three different markets only for energy trading: day-ahead market, intra-day market and centralized market for electricity bilateral contracts. Among these markets, the most competitive market is the day ahead market (spot market). The Day Ahead Market (DAM) is part of the electricity wholesale market where firm quantities of energy is traded for each trading interval (one hour) of the corresponding delivery day. The scopes of this market are to create a set up for a competitive, transparent and non-discriminative wholesale electricity market in Romania, reducing the trading prices for electricity and establishing reference prices for wholesale electricity market (www.opcom.ro).

RIVER OLT CASCADE OF HPP

The optimization algorithm was developed for the River Olt Hydroelectric Power Plant Cascade and more exactly for the middle sector. As show in Figure 1 the complex system of the cascade on River Olt consists of 30 low head hydropower plants (HPP) with 2 vertical Kaplan turbines per each, with a total capacity of 1.059 MW (after complete development) and will generate more than 2.8 TWh. The cascade is divided into three sectors: upper sector, middle sector and lower sector because of administrative reasons, grid regulations for dispatching reasons and balancing procedures. The middle sector begins with HPP Gura Lotrului and ends with HPP Arcesti and has a total capacity of approx. 555 MW. Each stage of the cascade has a dam, power house and auxiliary elements (connection station, transformation station, etc.). The lake for each power plant is different in terms of height, volumes (gross, net, etc.) and other utilizations of the water (e.g. water supply, industrial, irrigation, etc.). The installed discharge on every power plant is 330 m^3/s which in one hour means 1.188 x 10^6 m^3. The net volume of the lakes vary between 2,10 m^3 and 53,03 m^3 making it very difficult for most of the dams to keep the water for a longer period of hours and respect also the other technical conditions imposed by the operator.

Figure 1: River Olt Hydropower Cascade

HYDROPOWER PLANT OPTIMIZATION

The algorithms developed so far for the operation of River Olt Cascade founded in (Iana 2005) had two major approaches: maximizing the energy production and maximizing power production.

A new approach in the optimization of the operation of the cascade is proposed with a new objective: to maximize the generated income by taking into consideration that the entire energy production is traded on the DAM. The particularity of the DAM is that each trading interval (hour) has its own importance therefore the peak energy is the most expensive and the off-peak energy is a cheap energy. The prices for each hour are taken into consideration when planning the production for the next day. The proposals for DAM are given each day for the next day and the prices considered are estimated by statistical methods.

A comparison is presented between two different optimization functions with different objectives: objective 1 – maximizing the energy produced and objective 2 – maximizing the income (revenues generated). The purpose of this comparison is to show that the market behavior has an important influence on the production of energy and the energy has to be planned according to the market not necessary according to the biggest production. We can say that is better to produce less energy but expensive one, than to produce more energy but with less generated income due to the market principles.

MATHEMATICAL MODELLING

For the development of the mathematical model necessary for the optimization algorithm, the following sketch and notations taken from (Iana 2005) as shown in Figure 2.

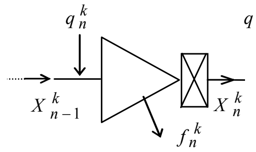


Figure 2: Notations used for cascade

The main parameters are q_n^k - affluent volume for hour k for the remaining hydrographic basin in lake n; X_n^k - effluent volume from lake n in hour k; f_n^k volume of water taken from the lake for other purposes (non-energetic). Furthermore, the volume account for each lake is presented in the following equation (1):

$$V_n^k = V_n^{k-1} + X_{n-1}^k + q_n^k - f_n^k - X_n^k \qquad (1)$$

A series of constrains were considered due to technical limitations and mathematical considerations: the used flow (with value between the minimum technical flow per turbine and maximum installed discharge per power plant), limitations in the speed of raising and lowering the level in the lake, availability of the equipment (e.g. both units available, one unit available or limitation in flow per unit or power plant). The effluent volume from each lake is made of two parts, the volume used for power generation T_n^k and the spilled volume D_n^k as shown in equation (2)

$$X_n^k = T_n^k + D_n^k \qquad (2)$$

For the energy production an equation wich depends on the level of the lake, the gross head and the net head of the power plant, and the used flow (volume) for energy generation as shown in (3):

$$E_n^k = a_n\left[b_n\left(V_n^{k-1} + V_n^k\right)^{c_n} - Z_{av} - d_n\left(\frac{X_n^k}{Q_n}\right)^2\right]\cdot T_n^k \qquad (3)$$

where the terms whithin parantesis estimates the net head of the power plant depending on the upper level, lower level and local losses through the power plant. The parameters a_n, b_n, c_n, d_n are constants determined for each lakes from the configuration of the dams and the capacity curves for lakes shown for example in Figure 3. The red chart represents the capacity curve between the minimum and the maximum allowed level. For the flexibility of the algorithm, the restrictions in the minimum and maximum levels were given according to the technical limits of the dam and the lake in order to give the possibility to the user of the algorithm to establish other restriction for the level compared to the specific requests (when interventions, dry season, water supply reserve, etc.).

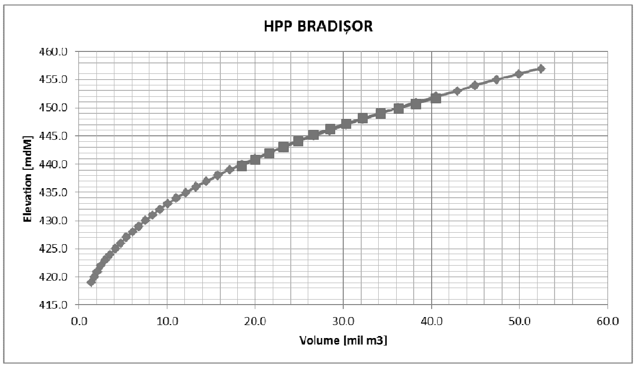


Figure 3: Capacity curve for HPP Bradisor

Maximizing the energy production (objective 1) of the power plant is written in equation (4):

$$\max\left\{ Energy = \sum_{n=1}^{N} E_n^k\left(\overline{V}_n^k, T_n^k\right) \right\} \qquad (4)$$

where energy is computed with equation (3).
In order to maximize the income generated by working under the DAM conditions and prices, another parameter was used - p_k, the **M**arket **C**losing **P**rice (MCP), determined by concurential and transparent methods for each trading hour by the Market Operator. The specificity of the DAM, as presented earlier, is shown is Figure 4 for a random selected day, used in the simulations.

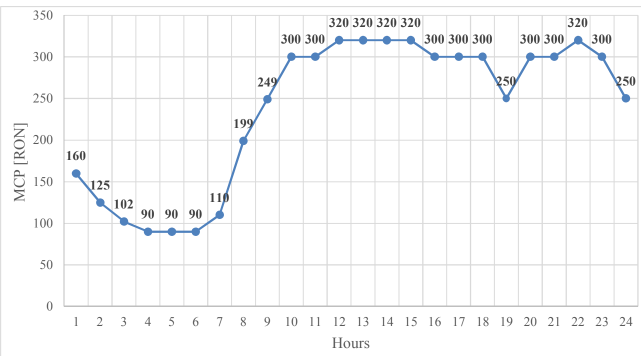


Figure 4: Market closing prices and volume traded on DAM

The equation (4) used for the optimization of the energy production is modified according to equation (5) with the pricing element from the Figure 4 in order to achieve objective 1:

$$\max\left\{ Income = \sum_{k=1}^{24} p_k \sum_{n=1}^{N} E_n^k\left(\overline{V}_n^k, T_n^k\right) \right\} \qquad (5)$$

SIMULATION USING GA

The GA optimization algorithm was developed under Pascal programming environment. The program has a set of GA related parameters that can be defined and selected in order to test the program ability to give an optimal solution to the problem. The parameters are the following: selection type (roulette wheel, tournament, normalized geometric sorting), mutation type (boundary, non-uniform) and crossover techniques. Results presented in this paper were obtained by using heuristic crossover, boundary mutation and normalized geometric sorting because with these parameters the best results were obtained. Other parameters of the algorithm are: mutation probability 0.03, crossover probability 0.8, maximum generation used 4000. When planning the energy production for the DAM, the time step of the simulation can be one hour ore more. The reason behind selecting a bigger time step is a technical one and it is related with aspects of reliability and maintenance complications that might occur when starting and stopping the hydro units. It is known that starting and stopping a machine (turbine and generator that have more than 300 tones combined) can cause to premature fatigue of the mechanical parts (Dragoi et al. 2013). Therefore, the simulations results presented in this paper consider a three hour time step for simulation. The decision structure of the cascade has the possibility to select different time steps but the experience of the dispatcher is to have a recommended time step of three hours. In this way, each unit will run at least three hours and when in stationary will stay again at least three hours. This means that the variables that represent the solution of the optimization problem are eight per each power plant, leading to a total number of variable of 120. Other elements of the algorithm are: the population dimension (180), maximum number of generations (24000), probability of crossover (80%) and probability of mutation (3%). The equations used were developed starting from the sketch of the entire middle sector cascade. An example is presented in Figure 5.

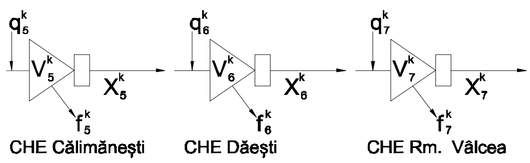


Figure 5: Example of sketch used for the cascade operation simulation

RESULTS

The results are presented for both maximization objectives and are based on the same conditions: availability of the units, same parameters of the lakes (levels in the lakes at the beginning and end of the day, hydrological regime, available flow, inflows in the lakes, installed discharges in the turbines with taking into consideration the minimum and maximum limitations, etc.) and the same cost of the energy per each trading interval as shown in Figure 4. The comparison is between two different hydrological regimes (dry season and above average season). A key condition was selected (but can be modified) that the volumes in the lakes at the beginning of the day are equal with the volumes at the end of the day. In order to satisfy this request, to the GA model was given a guideline to follow when simulation is run. Until the middle of the day (12:00) the target is to empty the lakes and lower the levels according to an interpolation between the two values, and after this point, the target is to fill the lakes in order to achieve the final value needed at the end of the day (24:00). Because the first power plant is linked upstream with another power plant with the same installed discharge the hydrological regime presented in Table 1 was selected for the flow in the first lake (time step selected of three hours):

Table 1: Hydrological regime at the first lake of cascade

Time step	1	2	3	4	5	6	7	8
Flows [m^3/s]	35	35	165	165	35	165	165	35

For the financial data of the DAM taken into consideration in Figure 4, the average price for the energy for the entire day is 238.15 RON/MWh. Having the regime presented in Table 2, the optimal policy according to the objective 1 generated an energy production of 9169.4 MWh with a gross income of 2461880 RON (Romanian Currency, equivalent of 547084 EUR) resulting a value of 248.49 RON/MWh. On the other hand, the optimal policy determined according to objective 2 generated an energy production of 8380.3 MWh but a gross income of 2543160 RON resulting a considerable higher value of 303.47 RON/MWh, as presented in Tabless 2 and 3, and in Figure 5.

Table 2: Results objective 1 – energy maximization

Time step	1	2	3	4	5	6	7	8	Total
Energy [MWh]	419.4	413.70	1,174.10	1,461.70	1,522.90	1,448.40	1,409.00	1,320.10	9,169.40
Income [RON]]	54.107	37.24	218.38	448.27	487.34	434.53	399.21	382.82	2,461.88
Income/MWh	129.01	90.01	186.00	306.68	320.01	300.00	283.33	289.99	268.49

Table 3: Results objective 2 - income maximization

Time step	1	2	3	4	5	6	7	8	Total
Energy [MWh]	7.40	0.00	211.10	2,377.00	2,639.60	1,945.10	332.60	867.30	8,380.30
Income [RON]]	0.96	0.00	39.27	728.95	844.67	583.54	94.25	251.53	2,543.16
Income/MWh	129.73	0.00	186.03	306.67	320.00	300.01	283.37	290.01	303.47

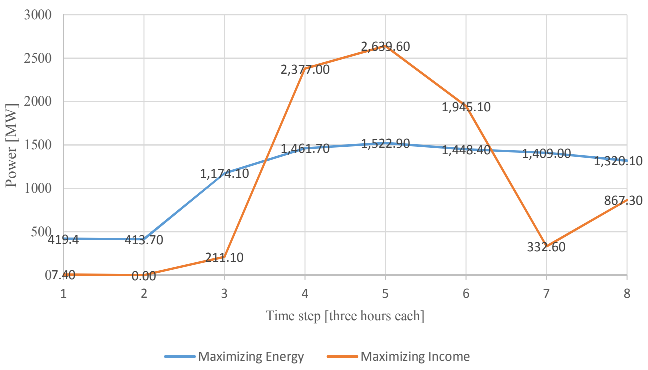


Figure 5: Power Generation distribution per day

For a second simulation, a different hydrological regime was selected and is presented in Table 4:

Table 4: Hydrological regime at the first lake of cascade

Time step	1	2	3	4	5	6	7	8
Flow[m^3/s]	190	195	185	185	195	195	195	190

The values presented in this table shows an average flow over the annual average flow which is around 165 m^3/s.
For the same financial data of the DAM taken into consideration (Figure 4), the optimal policy according to the objective 1 generated an energy production of 13060.3 MWh with a gross income of 3305270 RON, resulting a value of 253.08 RON/MWh. On the other hand, the optimal policy determined according to objective 2 generated an energy production of 12945.5 MWh but a gross income of 3700050 RON resulting a value of 285.82 RON/MWh, as presented in Tables 5 and 6, and Figure 6.

Table 5: Results objective 1 – energy maximization

Time step	1	2	3	4	5	6	7	8	Total
Energy [MWh]	1,055.50	1,225.00	1,468.20	1,597.40	1,783.70	1,873.40	2,019.00	2,038.10	13,060.30
Income [RON]]	136.16	110.25	273.08	489.86	570.79	562.03	572.06	591.05	3,305.27
Income/MWh	129.00	90.00	186.00	306.66	320.00	300.00	283.34	290.00	253.08

Table 6: Results objective 2 - income maximization

Time step	1	2	3	4	5	6	7	8	Total
Energy [MWh]	498.30	26.90	969.20	2,477.80	2,618.30	2,476.90	1,851.90	2,026.20	12,945.50
Income [RON]]	64.28	2.42	180.27	759.85	837.86	743.07	524.71	587.59	3,700.05
Income/MWh	129.00	90.00	186.00	306.66	320.00	300.00	283.34	290.00	285.82

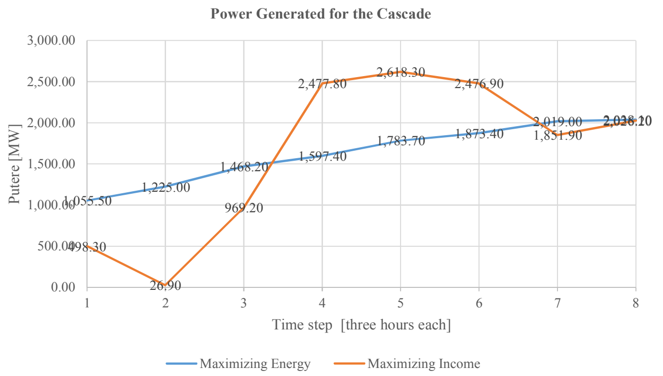


Figure 6: Power generation distribution per day

CONCLUSIONS

During flood season or above average hydrological season, the algorithm is not very competitive because of the limitation in the net volume of the lakes, which is relatively small compared to the installed discharge and the inflows, but still is offering a more feasible operation.
However, when the hydrological data used is according to a dry season, the algorithm has very good results in terms of optimal solutions.
The developed algorithm offers a powerful tool to determine the best optimal operation regime. The benefit from using the GA optimization offers an intensive search throughout the entire specter of available solutions and offers a local optimum to the problem which leads to incontestable economical results as presented in Tables 7, 8 and 9.
The energy generated is slightly smaller for objective 2 compared to the first. This is due to the variation of levels in

the lakes (smaller than in objective 1) which is leading to a smaller head, and therefore a limitation of power. However the objective 2 gives a solution where the energy production is prioritized according to the variation in MCP for the Day Ahead Market, making it suitable for a higher economic benefit.

Table 7: Energy production comparison

Case	Energy Production Objective 1 [MWh]	Energy Production Objective 2 [MWh]	Ratio [%]
I	9.169,40	8.380,3	-8.61%
II	13.060,00	12.945,5	-0.87%

Table 8: Generated income comparison

Case	Income Objective 1 [RON]	Income Objective 2 [RON]	Ratio [%]
I	2.461.880	2.543.160	+3.30%
II	3.305.270	3.700.050	+11.94%

Table 9: Generated specific income comparison

Case	Income/MWh Objective 1 [RON/MWh]	Income/MWh Objective 2 [RON/MWh]	Ratio [%]
I	268,49	303,47	+13.03%
II	253,08	285,81	+12.93%

REFERENCES

Banos, R.; Manzano-Agugliaro, F.; Montoya, F.G.; Gil, C.; Alcayde, A.; Gomez, J. (2011). "Optimization methods applied to renewable and sustainable energy: A review" *Renew. Sustain. Energy Rev.*, 15, 1753–1766.

Chau KW, Albermani F (2003), "Knowledge-based system on optimum design of liquid retaining structures with genetic algorithms". *J Structural Engineering* ASCE 129(10):1312–1321.

Chang, L.C.; Chang, F.J.; Wang, K.W.; Dai, S.Y. (2010). "Constrained genetic algorithms for optimizing multi-use reservoir operation". *J. Hydrology*. 2010, 390, 66–74.

Cheng, C.T; Wang, W.C; Xu, D.M; Chau, K.W; (2008). "Optimizing Hydropower Reservoir Operation Using Hybrid Genetic Algorithm and Chaos", *Water Resource Management,* 22:895–909

Chiu, Y.C.; Chang, L.C.; Chang, F.J. (2007). "Using a hybrid genetic algorithm-simulated annealing algorithm for fuzzy programming of reservoir operation". *Hydrology Process*., 21, 3162–3172.

Chootinan, P.; Chen, A. (2006). "Constraint handling in genetic algorithms using a gradient-based repair method:. *Comput. Oper. Res.* 2006, 33, 2263–2281.

Deb, K., (2000) "An efficient constraint handling method for genetic algorithms", 186, 311–338.

Dragoi, C., Piraianu V.F., Irimia P.C., (2013). "Lifetime evaluation of the mechanical elements for hydraulic turbines by numerical simulation", *Industrial Simulation Conferences, Eurosis*

Ferreira, A.R.; Teegavarapu, R.S.V. (2012). "Optimal and adaptive operation of a hydropower system with unit commitment and water quality constraints". *Water Resources Management*, 26, 707–732.

Holland, J.H. (2012)."Adaptation in Nature and Artificial System; The University of Michigan Press:Ann Arbor", MI, USA, 1975. *Energies*, 5 2910

Iana, Gh. (2005). "Optimization of hydroelectric power plant cascades with application to River Olt Cascade", PhD Thesis, University POLITEHNICA of Bucharest.

Labadie JW (2004) "Optimal operation of multi reservoir systems: State-of-the-art review". *J Water Resources Plan Manage* 130(2):93–111

Lin, C.Y.; Wu, W.H. (2004) "Self-organizing adaptive penalty strategy in constrained genetic search". *Struct. Multidiscip. Optim.*, 26, 417–428.

Mallipeddi, R.; Suganthan, P.N. (2010) "Ensemble of constraint handling techniques. IEEE Trans". *Evol. Comput*., 14, 561–579.

Nanakorn, P.; Meesomklin, K. (2001). "An adaptive penalty function in genetic algorithms for structural design optimization". *Comput. Struct.*, 79, 2527–2539.

Runarsson, T.P.; Yao, X. (2000). "Stochastic ranking for constrained evolutionary optimization". *IEEE Trans. Evol. Comput.*, 4, 284–294.

Wurbs RA (1993). "Reservoir-system simulation and optimization models". *J Water Resour Plan Manage.*, 119, (4):455–472

Xu, B; Zhong, P.A; Wan, X; Zhang, W; Chen, X; (2012) "Dynamic Feasible Region Genetic Algorithm for Optimal Operation of a Multi-Reservoir System", *Energies*, 5, 2894-2910

Yeh WW-G (1985). "Reservoir management and operation models: a state-of-the-art review". *Water Resource Res* 21(12):1797-1818

www.opcom.ro – Romania Gas and Electricity Market Operator

BIOGRAPHY

VLAD FLORIN PÎRĂIANU was born in Romania on the 4 of March, 1983. He received the degree in hydropower engineering from the University POLITEHNICA of Bucharest in 2007. He is a Ph.D. candidate in hydropower engineering within the same university, and he is working as assistant professor in the Hydraulics, Hydraulic Machinery and Environmental Engineering Department in the Power Engineering Faculty.

NICOLAE VASILIU graduated in Hydropower Engineering from University POLITEHNICA of Bucharest in 1969. He became a Ph.D. in Fluid Mechanics after a research stages in Ghent State University and Von Karman Institute from Bruxelles. He became state professor in 1994, leading the Energy & Environment Research Centre from the University POLITEHNICA of Bucharest. He managed five years the Innovation Romanian Agency. He worked always for the industry, as project manager or scientific advisor, promoting the numerical simulation as an engineering tool.

GHEORGHE IANA was born in Romania on the 11 of November, 1950. He graduated in Hydropower Engineering the University POLITEHNICA of Bucharest in 1974. He earned his PhD in 2005 with a thesis subject related to hydroelectric power plant optimization. He is the head of energy management and dispatching department in SC Hidroelectrica SA – Rm. Valcea branch.

VIDEO TECHNOLOGY

Adaptive Unequal Error Protection to Enhance Multicast Live Video Streaming

Ali Zanbouri[1], Marwan Al-Akaidi[2], Shakeel Ahmad[3]

[1]aelaywe@aou.edu.kw, [2]m.al-akaidi@arabou.edu.kw, [3]shakeel@dmu.ac.uk

KEYWORDS

Multicast video streaming, Unequal error protection, rateless code, forward error correction.

ABSTRACT

Unequal Error Protection (UEP) is an effective method to protect the video against packet losses over wireless packet erasure networks where the exact packet loss rate is hard to predict. A number of UEP schemes based on Rateless code have been proposed for video streaming. However, these schemes employ static UEP settings which do not adapt with the time varying packet loss rate of the channel. This paper proposes an adaptive UEP scheme for live multicast video streaming. The proposed scheme adapts the number of priority classes in the video stream and their protection levels dynamically according to the packet loss rates of all receivers in the multicast session. It also presents a heuristic based algorithm to adapt the UEP settings with changing packet loss rate. We conduct simulations by applying our proposed adaptive UEP scheme on H.264/SVC encoded video bit stream and show that adaptive UEP settings bring significant improvements in terms of average PSNR over a previously studied static UEP scheme for same used bandwidth.

I. INTRODUCTION

We are living in the century of accelerated and continuous events and news, therefore enhancing the quality of broadcast/multicast live video streaming over packet erasure wireless networks (3G, and 4G like Mobile WiMax and LTE) becomes more and more important topic nowadays.

In unicast video streaming one sender can send video stream to just one receiver while in broadcast/multicast video streaming one sender can send the same video stream to more than one receiver as shown in Fig. 1. For simplicity, in this paper we assume that one sender can send the same video to three heterogeneous receivers over three different packet erasure wireless channels.

Indeed, the on-demand and live video streaming techniques can tolerate some delay up to tens of seconds between requesting and playing out the video streaming. However, realtime video streaming applications (e.g., video conferencing) cannot accept delay more than 400 milliseconds.

Forward Error Correction (FEC) techniques have the potential to enhance the quality of live and realtime video streaming applications over packet erasure wireless channels. FEC can be based on fixed rate coding like Reed Solomon code or Rateless code like LT code [7] or Raptor code [8]. In Rateless coding the number of encoded symbols do not have to be fixed a priori and an infinite number of encoded symbols can be produced from a finite set of source symbols on the fly. Moreover, the encoding and decoding complexity of Rateless code is much less than many fixed rate FEC schemes. These properties make Rateless code an ideal choice in live streaming application over time varying channels where the packet loss rate can change abruptly.

In UEP scheme, the source symbols are categorized into a number of classes. For example in case of two classes using LT code, source symbols are divided into less important bits (LIB) and more important bits (MIB). Then the encoding process of LT code is modified such that the probability of selecting MIB symbols is set higher than that of LIB symbols. This leads to lower bits error Rate (BER) in MIB symbols as compared to that of LIB symbols [1]. Note that in equal error protection (EEP), all source symbols are selected with equal probability.

The authors in [2-4] proposed a robust UEP scheme based on LT code for video streaming. This scheme uses two classes of priority namely MIB and LIB. Further it uses Expansion Factor (EF) to increase the number of encoded symbols virtually to decrease the overall BER. UEP is achieved using Repeat Factor (RF) that increases the virtual length of MIB independent to that of LIB. This leads to lower BER in MIB as compared to that of LIB. We extend this scheme significantly for multicast streaming scenario where the number of classes and values of RF and EF are adapted dynamically according to the packet loss rate of all receivers.

We used H.264/AVC Extension for Scalable Video Coding (H.264/SVC) to generate a base layer and a number of enhancement layers. An enhancement layer can be decoded to improve the video quality only if all the previous layers have been decoded successfully. This dependency makes the base layer most important followed by higher layers with decreasing importance. We create classes of UEP from various layers of SVC bit stream. For example, in case of two classes, base layer can be placed in MIB and all enhancement layers can be placed in LIB.

We propose a heuristic based algorithm to adapt the value of EF, the number of priority classes and value of RF for each class dynamically according to the observed packet loss rate for each receiver. We conduct simulations and show that the PSNR averaged over all receivers is significantly better than that of previously studied UEP scheme with static UEP settings.

The rest of the paper is organized as follows. In Section II, we review the related work. Section III presents our proposed technique. Section IV discusses the experimental results. Finally Section V presents conclusions.

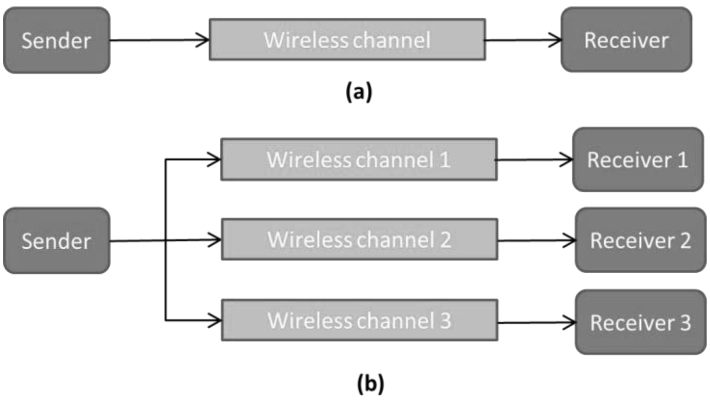


Fig. 1: (a) Unicast video streaming (b) Multicast video streaming

II. RELATED WORK

EEP uses the same level of protection for all source symbols while UEP assigns different levels of protection to different source symbols according to their importance in constructing the video streaming.

Rahnavard et al. [5] were the first to propose a method to provide UEP using LT code. Later on, Sejdinovic et al. [6] presented an UEP scheme which was shown to perform better than [5]. The UEP of Ahmad et al. [2-4] used a block duplication technique to achieve UEP using LT code. The scheme is shown in Fig. 2 with a toy example. In this toy example, there are 6 source symbols divided into two classes MIB and LIB. The first two source symbols belong to MIB while the next 4 symbols belong to LIB. The scheme uses Repeat Factor (RF) as the number of times source symbols belonging to a particular class are repeated. It uses Expansion Factor (EF) as the number of times all source symbols are repeated. This toy example uses RF=2 and EF=2 so that the original source block consisting of two MIB symbols and four LIB symbols is transformed into a virtual block of size EF(RFx2+4)=16. Higher RF means that the probability of selecting symbols from that class would be higher leading to lower BER. The information about the RF and EF is then conveyed to the receiver and used in the decoding process. For details on this scheme, we refer to [2-4].

All these schemes use static settings of UEP. The number of classes and the protection given to different classes are not updated dynamically with the channel conditions.

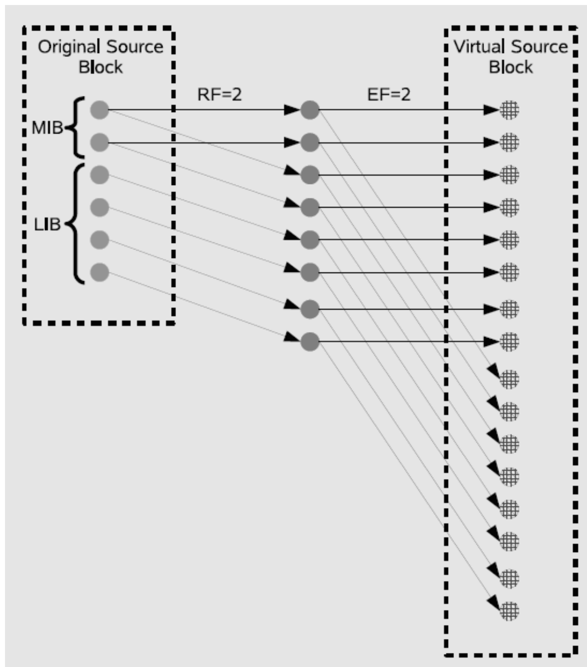


Fig. 2: Building a virtual source block of size 16 from the original source block of size (k) = 6, where EF = 2, and RF = 2 (Courtesy of [3]).

III. PROPOSED TECHNIQUE

In this section we describe our proposed adaptive UEP scheme to enhance the perceived video quality in broadcast/multicast live streaming under time varying packet erasure wireless channels. Figure 3 shows the block diagram of the proposed system for three multicast receivers.

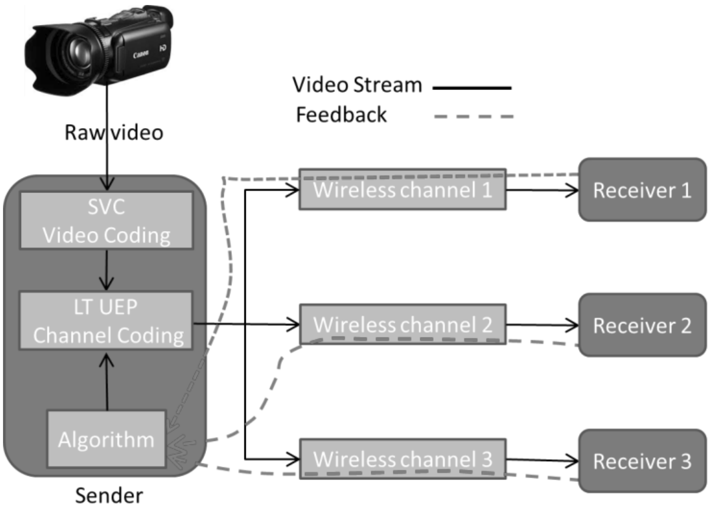


Fig. 3: The Proposed UEP scheme

The sender receives the raw video captured by the camera over a period T to create a video segment. This segment is encoded using H.264/SVC encoder to generate a base layer and a number of enhancement layers. The encoded video is sent to the LT UEP encoding unit that applies UEP according to the settings provided by the Algorithm. These settings consist of EF, number of classes and RF for each class. After applying UEP, the same video is transmitted to all receivers using UDP as transport protocol. Each receiver first applies LT decoding to recover base layer and enhancement layers and then decodes the video. In addition to that, each receiver observes the packet loss rate and reports it periodically to the Algorithm unit in the sender using the feedback channel. The algorithm uses packet loss rate from all receivers to calculate the value of EF, number of classes and value of RF for each class. In the following we give our heuristic based algorithm assuming temporal scalability.

1) If the average packet loss rate is less than 2%, then:
   a) The source symbols should be divided into three classes: MIB (I frames), Average Important Bits (AIB) (P frames) and LIB (B frames).
   b) Set RF to 3 for MIB, set RF to 2 for AIB and set RF to 1 for LIB symbols.
   c) Set EF to 20.
2) If the average packet loss rate is less between 2% and 15%, then:
   a) The source symbols should be divided into two classes: MIB (I and P frames) and LIB (B frames).
   b) Set RF to 14 for MIB and set RF to 1 for LIB.
   c) Set EF to 1.
3) If the average packet loss rate is greater than 15%, then:
   a) The source symbols should be divided into two classes: MIB (I and P frames) and LIB (B frames).
   b) Set RF to 20 for MIB and set RF to 1 for LIB.
   c) Set EF to 1 for all MIB and LIB symbols to achieve the final virtual source block.

Note that in all cases, the number of encoded symbols that are produced and transmitted are the same for a fair comparison. In the next section, we give experimental results.

IV. EXPERIMENTAL RESULTS

In our multicast scenario, we assumed three receivers but the proposed scheme can be generalized to any number of receivers. We used standard Stefan video sequence which has spatial resolution of 352x288 and a temporal resolution of 30 fps. The first group of pictures (GOP) of the sequence was encoded using the H.264/SVC reference software (JVSM) into one base layer (BL) and 14 enhancement layers. The GOP had a length of 16 frames. We used the same bit rate and layer sizes as used in [3].

Fig. 4 compares the average PSNR performance of our proposed adaptive UEP scheme with that of static UEP scheme in [3] for different settings of RF and EF. It can be seen that no static UEP setting gives better performance across the whole range of packet loss rate. Our scheme

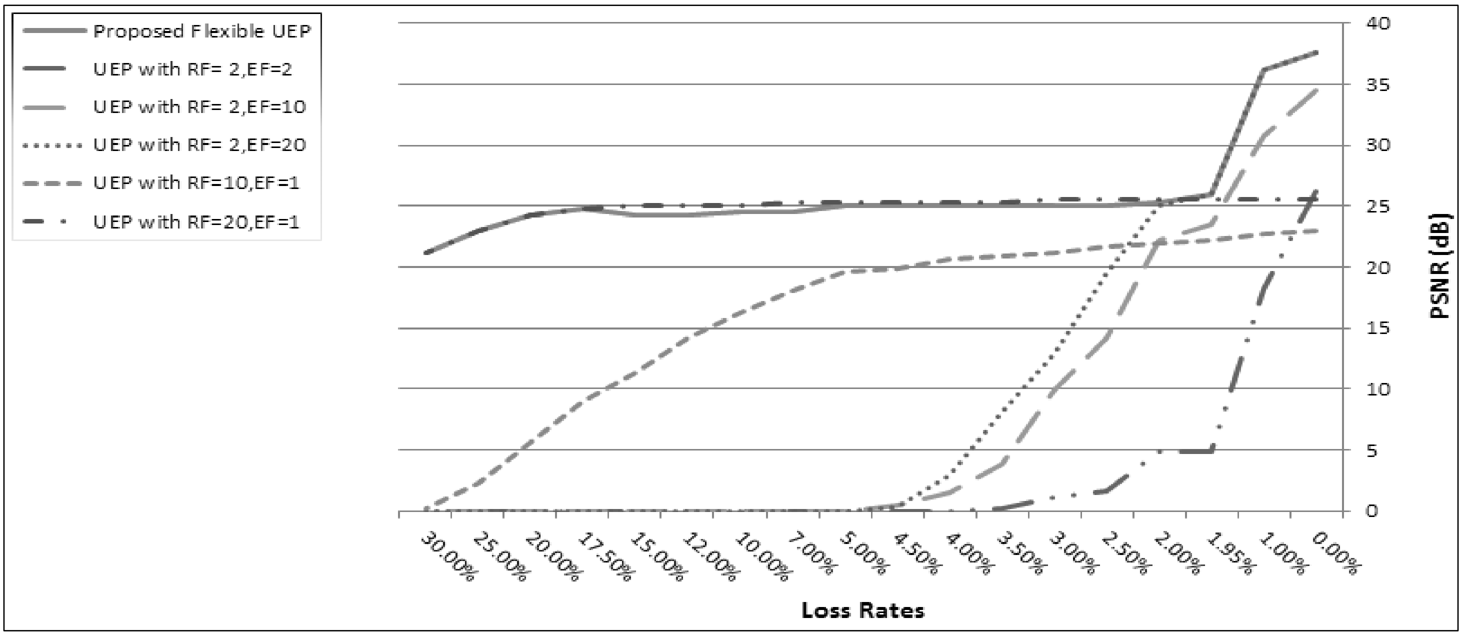


Fig.4: Average PSNR versus average observed packet loss rate using our proposed adaptive UEP and UEP scheme of [3]

since adapts its UEP settings dynamically with the current loss rate, it achieves better PSNR for the whole range of packet loss rates from 0% to 30%.

V. CONCLUSIONS

This paper has presented an adaptive UEP scheme for multicast video streaming. This scheme extends the previously proposed static UEP by adapting the UEP settings, i.e., the EF value, number of classes and RF value for each class dynamically based on observed packet loss rate averaged over all receivers in the multicast session. Experimental results using an H.264/SVC encoded video show that adaptive scheme outperforms the static scheme significantly in terms of average PSNR.

REFERENCES

[1] N. Rahnavard, H. Pishro-Nik, and F. Fekri, "Unequal Error Protection Using Partially-Regular LDPC Codes", IEEE Trans. on Communications, vol. 55, no. 3, pp. 387-391, March 2007.

[2] S. Ahmad, R. Hamzaoui, M. Al-Akaidi, "Unequal error protection using LT codes and block duplication", Proc. MESM 08, Middle Eastern Multiconference on Simulation and Modelling, Amman, Aug. 2008.

[3] S. Ahmad, R. Hamzaoui, M. Al-Akaidi, "Video multicast using unequal error protection with Luby Transform codes", Proc. MESM 09, Middle Eastern Multiconference on Simulation and Modelling, Beyrouth, Sep. 2009.

[4] S. Ahmad, R. Hamzaoui, M. Al-Akaidi, "Unequal error protection using Fountain codes with applications to video communication", IEEE Transactions on Multimedia, Vol. 13, pp. 92-101, Feb. 2011.

[5] N. Rahnavard, B. N. Vellambi, and F. Fekri, "Rateless codes with unequal error protection property", IEEE Trans. Inf. Theory, Vol. 53, No. 53, pp. 1521–1532, April 2007

[6] D. Sejdinovic, D. Vukobratovic, A. Doufexi, V. Senk, and R. Piechocki, "Expanding window fountain codes for unequal error protection", Proc. 41st Asilomar Conf., Pacific Grove, pp. 1020–1024, 2007.

[7] M. Luby, "LT-Codes", Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'02), 2002.

[8] A. Shokrollahi, "Raptor Codes", IEEE Trans. Inf. Theory, Vol. 52, No. 6, June 2006, pp. 2551–2567.

Linear Discriminant Classifier Ensemble for Face Recognition

H. A. Khan[1], Marwan Al-Akaidi[2]
Arab Open University,
Headquarters, Kuwait
h.khan@arabou.edu.kw [1]
m.al-akaidi@arabou.edu.kw [2]

Raja Waseem Anwar[3]
Arab Open University,
Oman Branch, Oman
waseem@aou.edu.om[3]

KEYWORDS
Face recognition, ensemble classification, linear discriminant classifiers, cross validation, wavelet analysis, principal component analysis.

ABSTRACT

Face recognition is an important field of research in a variety of domains including law enforcement, commerce, social media and marketing. We have developed a Linear Discriminant Analysis (LDA) based Ensemble Classification system for face recognition purposes. The proposed ensemble uses multi-resolution Wavelet analysis to extract features from two benchmark databases i.e. the ORL (AT&T) and the Yale image databases. Principal Component Analysis (PCA) has been employed for dimensionality reduction. The multi-resolution features are combined for efficient utilization for classification purposes. The ensemble has been trained using the Bagging technique and cross validated to obtain classification accuracy of 98.27% for the ORL database and 95.15% for the Yale database.

1. INTRODUCTION

Face recognition is an important area of research and finds wide application in law enforcement, commercial and social media networking domains. In the law enforcement domain, CCTV systems have become integral part of public safety and surveillance. Face recognition is one of the key elements available to law enforcement agencies to process cases where breaches of law are alleged. CCTV cameras are also available in commercial centers and the images captured by them could be used for marketing and product offering to potential customers.

Image capturing systems have experienced great proliferation in recent years resulting in large number of social networks making heavy use of images such as Instagram, Pinterest and Snapchat etc. Smart phones with powerful cameras are now available to the masses and the amount of images being captures is phenomenal. Along with the camera technology the storage and computational power is now commonly available to store and process large amount of images using cloud computing. Imaging systems are also available now as wearable such as the Google Glass Project. Face recognition is important for searching large image databases of social networks.

The structure of the paper is given below. Overview about face recognition techniques is given in section 2 of this paper. Section 3 discusses the basic ensemble methodology used in classification tasks. Information about the image databases and the experiments is provided in Section 4. Section 5 contains discussion about the feature extraction method used in our work. Transformation of features and their combination is presented in Section 6. Section 7 discusses the Bagging technique that has been adopted to train the ensemble of classifiers. Results have been discussed in Section 8, conclusions have been presented in Section 9 and future work suggestions are given in Section 10.

2. OVERVIEW OF FACE RECOGNITION TECHNIQUES

Researchers have applied a wide variety of techniques to explore the face recognition problem. An excellent survey about face recognition methods has been provided by (Zhao et al. 2003). The face recognition homepage also provides useful information about face recognition (Face recognition homepage). A popular methodology used for face recognition is the subspace method (Turk and Pentland 1991; Belhumeur et al. 1997; Gregory S. and Baback M. 2011). Pioneering work on subspace methods include the work on Eigenfaces for image analysis by (Turk and Pentland 1991). Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Independent Component Analysis (ICA) are popular techniques for face recognition. Kernel methods have also been proposed for face recognition purposes (J. Lu et al. 2003).

Face recognition methods are also based on the Elastic Bunch Graph Matching (EBGM) methodology (Wiskott et al. 1997). The EBGM methodology exploits the topological structure of the human face. Apart from 2-D analysis of faces, 3-D face recognition techniques have also been studied by researchers (A. Bronstein et al. 2004). The 3-D analysis of face images is more computationally intensive. Ensemble based classifier techniques have also been used for face recognition applications and have been studied by (El-Bashir 2012; J. Lu et al. 2006). Bagging technique has been used in (Ebrahimpour 2007) and a correct classification percentage (CCP) of 97.5% has been achieved. These ensemble techniques utilize the power of multiple classifiers in order to achieve classifications. Some other techniques employed for face recognition include the SVM, HMM and the Trace Transform (Face recognition homepage).

3. ENSEMBLE ARCHITECTURE

A basic ensemble architecture for classification purposes is shown in Figure 1. The ensemble consists of a number of individual classifiers who work together to achieve classifications results. For face recognition tasks, the main input to the ensemble classifier consists of the database of faces images. Pre-processing can be performed before the feature extraction step. The extracted features are fed to multiple base classifiers for processing. The final classification result is computed by the decision block.

In our work we have used the ClassificationEnsemble class of Matlab with Linear Discriminant classifiers as base classifiers in the ensemble system. A set of 30 base classifiers constitute the ensemble. The feature extraction block is based on the 2-D discrete wavelet decomposition. Experiments have been performed on two benchmark face image databases i.e. the AT&T Database of faces (AT&T Laboratories Cambridge), formerly known as the Olevetti Research Lab (ORL) Database of faces, and the Yale face image database (Yale face image database).

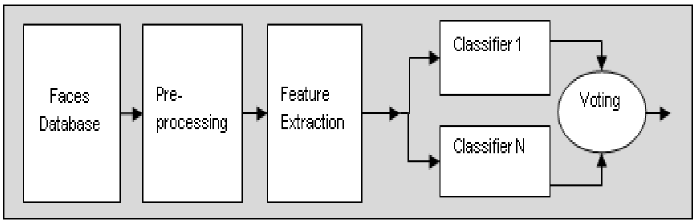


Figure 1: Ensemble architecture for face recognition system

4. IMAGE DATABASES AND SYSTEM SETUP

As stated above, we have considered two face image databases i.e. the ORL and the Yale Database of faces for face recognition purposes. The ORL database consists of 40 subjects with 10 images for each subject. The Yale databases consists of 15 subject classes with 11 images each. Snapshots from the two face image databases are shown in Figure 2 and 3.The snapshot from the Yale database clearly shows the pose, illumination and expression variability. Hence the Yale is a challenging database for face recognition since the subjects exhibit different exposure to varying light conditions and varying facial expressions. Relevant information about the databases and the system setup has been provided in Table 1 below:

Table 1: System Setup Information

Item/Factor	ORL	Yale
Total subjects in Database	40	15
Number of images of each subject	10	11
Total Number of Images	400	165
Type of Base Classifiers	Linear Discriminant	Linear Discriminant
Number of base classifiers	30	30
Testing/Validation methodologies	10-Fold	10-Fold
Image resolutions	Size 112X92	Size 64x64
Feature extraction methodology	2-D DWT	2-D DWT
Wavelets used	db2	db2
Wavelet decomposition levels	Level 1, 2, 3	Level 1, 2
Dimensionality reduction by	PCA	PCA

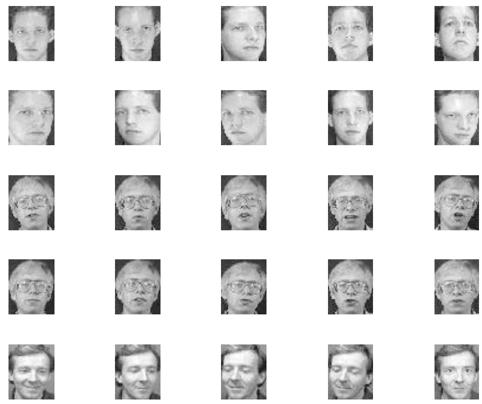


Figure 2: A snapshot of the images from ORL database (AT&T Laboratories Cambridge)

Figure 3: A snapshot of the images from Yale database

5. FEATURE EXTRACTION

The feature extraction block is one of the major blocks of the classification system. We use multi resolution analysis by employing the 2-D Discrete Wavelet Transform (DWT) to extract features from the face images. The Wavelet Transform is a well established technique for multi-resolution and multi-scale analysis of 1-D and 2-D signals (Daubechies 1992). The 1-D Wavelet Transform of a continuous time signal x(t) is given by (Mertins 1999):

$$W_x(b,a) = |a|^{\frac{-1}{2}} \int_{-\infty}^{+\infty} x(t)\psi^*\left(\frac{t-b}{a}\right) dt. \qquad (1)$$

The above shows that the Wavelet Transform is the inner product of the signal x(t) with a translated and scaled version of the wavelet $\psi(t)$.The parameters 'a' and 'b' are the scaling and translation parameters respectively.

We have used 3 levels of decomposition for the higher resolution ORL database images and 2 levels of decomposition for the lower resolution (size 64x 64) Yale database. We have employed a pre-processing step of histogram equalization for the Yale database images for improved performance. The DWT decomposition results in both the Detailed and the Approximation Coefficients. We use the Approximation Coefficients as feature vectors since they carry useful information about the images as depicted in Figure 4 below.

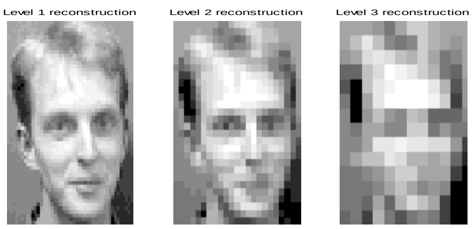


Figure 4: Reconstruction using approximation coefficients

6. TRANSFORMATION OF FEATURES AND THEIR COMBINATION

The various decompositions of the images by the DWT lead to high dimensional feature vectors. Using high dimensional feature vectors for classification purposes often leads to poor performance of classifiers. Hence it is desirable to reduce the dimensionality of feature vectors to reduce computational cost as well as to achieve better performance. We utilize the PCA technique for dimensionality reduction of the individual feature vectors. The reduced dimensionality feature vectors are then concatenated to achieve the final feature vector which is used for classification purposes. The concatenation of different decomposition level feature vectors results in better performance of the ensemble classifier since it utilizes the combined information contained in the different decomposition levels.

7. BAGGING TECHNIQUE FOR ENSEMBLE TRAINING

As mentioned above, the ensemble system used in this work uses the Linear Discriminant analysis based classifiers as base classifiers. The ensemble systems are commonly trained using the Boosting or the Bagging methodologies for ensemble learning. In this work the ensemble has been trained on the face image features data using the Bagging Technique for ensemble learning. Bagging or "bootstrap aggregation" uses resampled version of features data to train the ensemble (Matlab Statistics Toolbox). The cross validation methodology has been used to assess the performance of the ensemble system on the two benchmark databases.

8. RESULTS AND DISCUSSIONS

8.1 ORL Database Results

The proposed ensemble system has been implemented in the Matlab programing environment. The performance of the ensemble on the ORL database is shown in Figure 5. Three levels of DWT decomposition are achieved using the Daubechies 2 (db2) family of wavelets. The dimensionality of the feature vectors is reduced using the PCA technique. The ensemble computes the 10-fold cross validation error using individual decomposition level feature vectors as well as the response to concatenated or combined feature vector. The ensemble performance is computed by progressively varying the number of classifiers in the ensemble from 1 to 30. The experiments have been conducted for 10 runs and then the average performance has been computed. The Figure 4 shows that the cross validation error reduces as the number of classifiers in the ensemble increases. The figure also shows that the concatenated or combined feature set has the best performance. Table 2 shows that a classification accuracy of 98.27% was achieved by the ensemble for the ORL database.

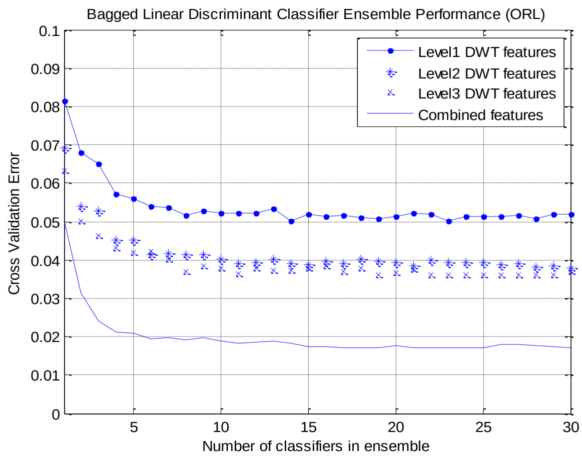


Figure 5: ORL database results

Table 2: Classification Accuracy for Ensemble (30 base classifiers) on ORL database

Feature Type	Classification Accuracy (%)
Level 1 DWT	94.80
Level 2 DWT	96.22
Level 3 DWT	96.30
Combined	98.27

8.2 Yale Database Results

The results for the Yale Database have been obtained using the same system setup as described above. Since the images in this database are lower resolution therefore we have used two levels of DWT decompositions in this case. Figure 6 shows the performance of the ensemble classifier on the Yale database. Since the Yale database images are of lower resolution (size 64x64) hence these are more challenging to classify. Also, the Yale database images have more pose, intensity and expression variations compared to the ORL database. This makes the task of classifying the images in this database more challenging compared to the ORL database. It is clear from Figure 6 that the concatenated or combined feature set resulted in best performance with Classification Accuracy of 95.15% (Table 3). Although the classification accuracy is lower compared to the ORL database it is comparable keeping the challenging nature of this database in mind.

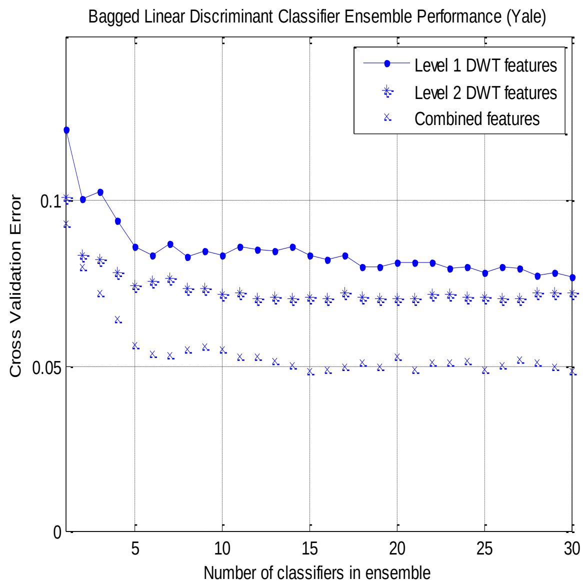


Figure 6: Yale database results

Table 3: Classification Accuracy for Ensemble (30 base classifiers) on Yale database

Feature Type	Classification Accuracy (%)
Level 1 DWT	92.30
Level 2 DWT	92.79
Combined	95.15

9. CONCLUSIONS

We have obtained classification accuracy of 98.27% for the ORL database and 95.15% for the more challenging Yale database using LDA based classification ensemble of 30 base classifiers and 10-fold cross validation. Feature vectors have been extracted from face images using the DWT decomposition. Dimensionality reduction of feature vectors has been achieved using the PCA technique. The results achieved in this work have demonstrated the effectiveness of the proposed Linear Discriminant Classifier Ensemble. It has been shown that the combined feature vectors resulted in better ensemble performance compared to individual feature vectors.

10. FUTURE WORK

The performance of the ensemble classification system is influenced by a number of factors such as the type of base classifiers, the number of base classifiers, the ensemble learning technique, type of features, dimensionality of feature space etc. In future we plan to consider other learning schemes for the ensemble system training. Also, future work would consider other types of base classifiers as members of the ensemble.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the research support provided by the Arab Open University (AOU).

REFERENCES

A. Bronstein, M. Bronstein, R. Kimmel, and A. Spira 2004. “3D face recognition without facial surface reconstruction”. Proceedings of ECCV 2004, Prague, Czech Republic, May 11-14.

Belhumeur P., J. Hespanha and D. Kriegman. 1997. “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, pp. 711-720.

Daubechies I. 1992. “Ten Lectures on Wavelets”, SIAM.

Daugman J. 2003. “Demodulation by complex-valued wavelets for stochastic pattern recognition”.

International Journal of Wavelets, Multiresolution and Information Processing, Vol. 1, No. 1, pp. 1-17.
Ebrahimpour H. and Kouzani A. 2007. "Face Recognition Using Bagging KNN". International Conference on Signal Processing and Communication Systems, Gold Coast, Australia.
El-Bashir M. S. 2012. "Face Recognition Using Multi-Classifier". Applied Mathematical Science, Vol. 6, no. 45, 2235-2244.
Garcia C., G. Zikos and G. Tziritas. 2000. "Wavelet packet analysis for face recognition". Image and Vision Computing, Vol. 18, No. 4, pp.289-297.
Gregory S. and Baback M. 2011. "Face Recognition in Subspaces". Handbook of Face Recognition, pp. 19-49.
Heseltine T. D. 2005. "Face Recognition: Two-Dimensional and Three-Dimensional Techniques". PhD thesis, The University of York, Department of Computer Science.
Jain A. K., A. Ross, and S. Prabhakar. 2004. "An introduction to biometric recognition". IEEE Transactions on circuits and systems for video technology, vol. 14, no. 1, pp. 4–20.
J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, S.Z. Li 2006. "Ensemble-based Discriminant Learning with Boosting for Face Recognition". IEEE Transactions on Neural Networks, Vol. 17, No. 1, pp. 166-178.
J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos 2003. "Face Recognition Using Kernel Direct Discriminant Analysis Algorithms". IEEE Trans. on Neural Networks, Vol. 14, No. 1, pp. 117-126.
Kirby M and L. Sirovich. 1990. "Application of the Karhunen–Loeve procedure for the characterization of human faces". IEEE Trans Pattern Analysis and Machine Intelligence, 12:103 108.
Mertins A. 1999. "Signal Analysis: Wavelets, Filter banks, Time-Frequency Transforms and Applications", John Wiley.
Turk M. and A. Pentland. 1991. "Eigenfaces for Recognition". Journal of Cognitive Neuroscience, Vol. 3, No. 1, pp. 71-86.
Wiskott L., J. Fellous, N. Kruger and C. Von Malsburg. 1997. "Face Recognition by Elastic Bunch Graph Matching". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, pp. 775-779.
Zhao W., R. Chellappa, A. Rosenfeld and P.J. Phillips. 2003. "Face Recognition: A Literature Survey". ACM Computing Surveys, pp. 399-458.
W. Zhao and R. Chellappa. 2002. "Image-based Face Recognition: Issues and Methods". Image Recognition and Classification, Ed. B. Javidi, M. Dekker, pp. 375-402.

WEB REFERENCES

AT&T Laboratories Cambridge. "Database of Faces, (formerly 'The ORL Database of Faces')". http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.
Face recognition homepage. http://www.face-rec.org.
Yale face image database. http://vision.ucsd.edu/content/yale-face-database.
Matlab Statistics Toolbox. http://www.mathworks.com/help/stats/index.html.

GAMING

A Framework For Adaptivity In Educational Serious Games Based On Agent Systems

Ahmed Elsherbiny Hasan
Damir Ismailovic
Bernd Brügge
Department of Computer Science
Technische Universität München
Boltzmannstr 3, 85747 Garching bei München,
Germany

KEYWORDS
Adaptivity, Serious games, Learning, Artificial Intelligence, Intelligent Agents.

ABSTRACT

Serious games can be used in an educational context to create a learning situation that entertains learners and let them enjoy their learning experience. In order to keep the joy and entertainment of (non-serious) games, game designers have to keep their eyes on the motivation of the player and the challenge the players are facing while they are playing the game. Motivation and challenge are driving forces behind learning. Learners will not be willing to learn (or at least without using their full potential) if they could not find suitable motivation and challenge. To achieve a reasonable amount of challenge so as to keep learners motivated, we need to adapt the game according to the player's skills and preferences. Adaptive educational games can be found, but still there is a lack of a generalized adaptivity framework. We present a literature review and general approach that can be used as a framework to add adaptivity to serious games using Intelligent Agents that can use techniques of Artificial Intelligence (AI) and Machine Learning (ML) in implementing adaptive games. We perform a usability evaluation on the user interface to ensure that it meets users' expectations.

INTRODUCTION

A serious game (SG) is a game that aims mainly for purposes other than pure entertainment. Serious games have been increasingly popular; They attract the interest of researchers and professionals in various fields such as education, health, vocational training, governance and defense. The serious game sector is expected to grow significantly in the medium term. IDATE Consulting and Research estimated that the serious game business generated revenue of 1.5 billion Euros around the world in 2010, and with an annual growth of about 47% between 2010 and 2015, is predicted to reach 10 billion euros at end of this period (Michaud 2010). The United States military has already started using serious games in promoting its culture, helping in the recruiting process and in training soldiers. Examples include the development of America's Army (Nieborg 2004) and "Tactical Iraqi Language and Culture Training System (TILCTS)" (Johnson et al. 2005) which is an educational serious game for teaching Iraqi Arabic Language and Iraqi culture. The later was a sucess story with 75% of trainees and officers agreeing that it was a useful training tool (Yates 2007). Additionally, serious games can be used in different kind of real world applications and problems. People enjoy solving problems, especially in context of learning and education, while playing games (Susi et al. 2007). Serious games have already proven their effectiveness, as they can provide a deeply immersive experience that increases engagement and motivation to reach the learning goal (Quinn and Neal 2008).

In order to keep the game fun and challenging, games need to adapt their challenge level automatically according to the player skills. Additionally, dynamically increasing the difficulty of the game according to player performance can help improving player's skills. That is why adaptive games that adapts to individual user's skill are essential for learning. Adaptive games are difficult to develop, nevertheless they provide more fun and enrich the player experience They can also reduce development effort because the game will adapt itself, so the developer does not need to consider all possible situations while implementation (Ram et al. 2007).

The aim of this paper is a literature review for adaptivity in serious games and to present a framework to help developers to add Adaptivity in serious games using Intelligent Agent (IA) system. The framework will also provide the ability for teachers and parents to track development of the students' skill. Many approaches of adaptive games depend heavily on the availability of domain experts for the target domain; therefore we discuss how we use Intelligent Agents to play the role of domain expert by implementing adaptivity. The Intelligent Agent in our framework will act as an Artificial Intelligence (AI) tool box that contains algorithms to adapt games, where the agent can be developed to use different algorithms for Adaptivity. Our framework allows a developer to use different AI algorithms that match the required task. In order to achieve the functionality of adaptive game, our framework must evaluate the skills of the player. This evaluation will be done by monitoring and analyzing the player's actions, and sending this information to the Intelligent Agent to analyze, and to take appropriate steps to adapt the game.

THEORETICAL BACKGROUND

Educational serious games, or for short, educational games, are serious games with a purpose of teaching players a set of pre-defined learning goals. Educational games are the most obvious and popular form of serious games (Ulicsak 2010). Taking into consideration (Vygotsky 1978) argument that playing is first form of learning and (Rieber 1996) argument that playing is an effective mediator for learning indicates the potential effectiveness of educational games. Educational serious games can be used in class rooms or as auxiliary materials out of class. They can also be used to teach concepts which are not in certain classroom curriculum, and in teaching adults. One of main advantages of educational serious games is that they provide a tailored learning experience that can use person's strengths, and work on or avoid their weaknesses. Computers and mobile devices are widely available everywhere, which increase the learning opportunities for students and learners, and can speed up the learning process. These educational games can help also in preventing developing false mental models about learning content (Haladjian et al. 2012).

It's essential for designing educational game to understand design and learning theories (Harteveld et al. 2010). According to Piaget Learning Theory, the learning takes place only when a person is in a state of disequilibrium (i.e., mental structures not in balance). Learning is happening as the construction of new knowledge resulting from the resolution to the conflict (Rieber 1996), which is a common situation in games. We will refer to some theories which will be used in defining adaptivity in serious games later.

The first theory we will examine is Knowledge Space Theory (KST), a mathematical model introduced by (Doignon and Falmagne 1985) which can be used to characterize learners based on their skills, and allow experts to represent domain knowledge based on skills which the learner needs to acquire to learn certain domain. KST represent the relation between skills in precedence order. It shows the learner needs to master certain skill before being able to move to the next skill. Figure 1 shows an example of a precedence diagram for some basic mathematical skills. So, to be able to get skill of adding, the learner has to master counting and recognizing numbers. Therefore if the player shows evidence of an ability to add two numbers, we can assume that this player must have the skills of counting and number recognition. In KST, a Knowledge State is defined to describe the learner’s state of knowledge as a set of skills that the learner has mastered.

The second concept we need to clarify is the Zone of Proximal Development (ZPD) which is introduced by (Vygotsky 1978). This zone is the difference between what the learner can do alone and without help, and what he or she can do with the help of a tutor or peer who has more knowledge and understanding of the topic. This person is called More Knowledgeable Other (MKO). Vygotsky believes that learning process takes place in this Zone. A learner's individual ZPD can define the minimum and maximum difficulty of the learning content that the learner can understand with external help. So, keeping an eye on the learner's individual ZPD is vital in an adaptivity process.

Lastly, Social Development Theory (SDT) is another theory developed by Vygotsky. SDT suggesting that (a) social interaction plays a fundamental role in development of cognition (and so to learning process) (b) the More knowledgeable other (MKO) is guiding the learner through the learning process, (c) and this learning process occurs in Zone of Proximal Development (ZPD).

Another important aspect that affects learning is what called learning style defined in (Kolb and Kolb 2005), which demonstrate differences between learners based on how they acquire knowledge more efficiently. This factor will be effective while designing adaptivity; if the learner can get instruction modality which is adapted for their learning style, the learning outcome is assumed to be better (Bellotti et al. 2009). Learning style grid can be found in (Kolb and Kolb 2005)

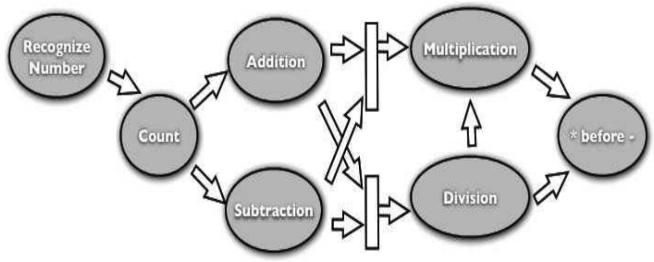


Figure 1: Example of Precedence Diagram (Ismailovic 2012)

ADAPTIVITY

Adaptivity means that the system automatically adapt to the user needs, sometimes called online adaptation. Adaptivity is shaping itself as a rapidly maturing field in games with significant developments both in industry and academia. These developments not only show good results in adapting the challenge level for a player, but also impact other affective states like fun, frustration, predictability, anxiety or boredom (Lopes and Bidarra 2011). There are two approaches to adapt serious games; either externally by experts to guide adaptation, or through the use of predefined levels. However, this is often not effective because the cost of an expert to guide adaptation process, while at the same time predefined levels solution might not fit all learners and adds the possibility that users will mis-estimate their personal skill level. Therefore, (automatic) adaptivity techniques can be employed to optimize these problems (Westra et al. 2008)

Adaptivity in the context of educational serious games is defined according to (Ismailovic et al. 2012) as “*an approach that enables a serious game to (A) learn from learner's behavior by (A1) intelligently monitoring and (A2) interpreting learner's actions in the game's world and (B) to intervene in the game by (B1) automatically adjusting the learning content and (B2) the game elements according to (C) the student's individual ZPD [Zone of Proximal Development] as neccessary and using the principles of (D) MKO [the More Knowledgeable Other], where ADAPTIVITY is a MKO for the learner according to the SDT [Social Development Theory]*“. In (Vygotsky 1978), he argues that the help of human tutors in learning process is maximized by working closely with learners and observing their actions and outcomes. Then, using the information the tutors have observed, they can adapt to the individual skills and needs of learners. This adaptive behavior is also desired for serious games. Because in the current state of classes it's not possible to provide a tutor for each student, an Adaptivity system can take the role of personal tutor (Ismailovic et al. 2012). In order to implement adaptivity we have two approaches suggested by (Manslov 2002): indirect adaptation and direct adaptation. By applying indirect adaptation, the adaptation occurs by extracting statistics from the game world that are used by conventional AI layer to adapt an agent's behavior, and the role of the learning mechanism is restricted to collecting information, thus playing no direct rule in changing agent behavior. Only the logic of Agent behavior will lead to the adaptation according to the new state of collected information. In contrast, the direct adaptation, involves applying learning techniques directly in the Agent's behavior itself. The indirect approach has advantages that (1) collecting information from the game world usually is an easy and reliable task which leads to more effective adaptation and (2) the change of behavior is centralized in the AI layer thus easy to debug and test. The indirect approach has one main disadvantage that it requires both information used while learning and the change of behavior to be defined by the AI designer. The advantages of direct approach is maximized when there little information about how the game should adapt. In our framework we used indirect approach. Key points of sucess in adaptation are (1) collecting as much knowledge as possible and (2) defining a good performance measure (Manslov 2002).

Motivation is a major drive in learning and playing. The advantage of serious games over other learning media is the fact that a serious game challenges the learner and keep him or her motivated during the learning process (Quinn and Neal 2008) and can also provide immersive engaging learning experience (Peirce et al. 2008). At the same time, having the same game for all learner will possibly decrease motivation and enjoyment. The challenge is to match individual skills and personality to reach highly desirable flow state (Westra et al. 2008). Flow is the mental state of operation in which a person performing an activity is fully immersed in a feeling of energized focus, full involvement, and enjoyment in the process of the activity (Csikszentmihalyi 1990). Flow can also be defined as a channel between anxiety and boredom that relies on the relation between the challenge of a task and individual's skill level. If a learner’s skill is high and he or she is faced with little challenge, the learner usually feel bored, and if skills is low but challenge is high, the person will feel anxiety and will not enjoy the experience. In case of learning situation where skills increase over time, the learning experience should provide higher challenge to keep learner in the flow channel.

An Intelligent Agent (IA) system, or simply Agent System, uses an autonomous entity (agent), possibly a software component, that observes the environment through sensors and acts on the environment using actuators according to the analysis of observations and internal state of the agent so as to achieve certain goals of the agent. Agents can be complex and can possibly learn or use knowledge, models and utility functions in the process of making decision. In our framework, the Agent is a model of a personal tutor or domain expert that adapt the learning content according to the player/learner performance and skill level. With a centralized service for adaptivity, it becomes difficult to scale increasing game complexity. Using Agents, the level of game challenge can more easily be scaled while keeping central control, therefore multi-agent organizations are being increasingly used in serious games adaptivity (Westra et al. 2011). Using agents in adaptive games is associated with some benefits; Simple game structure, because agents will allow characters of the game world to be autonomous and able to take their individual decisions that simplify game structure; less complex management and control, as an agent-based approach will allow distributed control of the game; easier real-time interaction by taking advantage of agents' properties like autonomy, reactivity and pro-activity; and finally, mediation of adaptivity in serious games by using techniques of Artificial intelligence to implement and adapt agent behavior (Hocine and Gouaich 2011).

Applications of Artificial intelligence in the introduction of adaptivity in serious games can be employed in different aspects such as user identification and content adaptation. The user identification task can be divided into two sub tasks: modeling the user with respect to interactive learning, and detecting user engagement and motivation. Content adaptation is the task of presenting a personalized view of learning material to the player which can be divided into Content Personalization to Learner, and adaptation to User Experience and Learning Goals. In order to realize content adaptation, different strategies can be used: these include Interactive Storytelling, Procedural Level Generation and Adaptive Game Balancing (Brisson et al. 2012). Interactive storytelling is a way in which the user can influence the progression of the story. There are two variations: a character centered approach; where each character has its own AI module that character uses to evolve the story, and an author-centered approach, where a central Agent act as the author of the story, and decides what will happen next (Olaverri Monreal et al. 2013; Brisson et al. 2012). Procedural Level Generation is a process of generating content tailored to the user characteristics; the tailored content can be content related to the game world and its elements, or learning material included as exercises or training scenarios (Lopes and Bidarra 2011). Adaptive Game Balancing adapts game features such as the challenge level to match player's performance.

RELATED WORK

The first project we want to present is the Dynamically Adaptive Serious Games (DASG) Framework (Ismailovic et al. 2012). DASG is a framework used in developing adaptive serious games. It has been used to develop different serious games in Technische Universität München (TUM). The framework consists of two parts: an adaptivity meta-model, and processes for developing serious games; namely a Domain model, a Monitoring meta-model, a Learner model and a Learning process model. In DASG the educational domain is represented as skills. The Domain model describes learning skills and skills necessary to play the game. The Monitoring meta-model can be used to monitor events in serious games. Learner Model represents the learner as a set of accquired skills, and with a profile that contain a history of actions which the learner executed in the game to accquire skills. The Learning process model defines the learning process in a serious games that contains playing or learning situations and assessment situations. It defines also how different situations in the game can show the performance of the learner.

The Hamlet system (Hunicke and Chapman 2004) is primarily a set of libraries embedded in the Half Life game engine to implement adaptivity. The main functions of Hamlet are (a) Monitoring game statistics according to predefined metrics (b) Defining adjustment action and policies (c) Executing actions and policies (d) Displaying data and system control settings (e) Generating traces and logs of play sessions. When player is moving around the game world, the Hamlet system monitors the player's actions so that, over time, it can predict what the player is going to do, and intervene by changing the game settings when it predicts that the player is moving to a state he or she should avoid. Some interesting results are shown, in that most players didn’t recognize that the system was helping them, as well as reporting slightly higher levels of enjoyment for expert players.

SOLUTION

According to (Ismailovic et al. 2012), an adaptive system should be able to describe the learning domain, monitor and chracterize the learner and intervene in the game based on the collected data. We have added that the framework should show the learner’s skill level. We have developed a solution to simplify the process of making a serious game adaptive that is extensible by adding more skills, learning goals, AI techniques and game elements.

We have used a client/server architectural style to increase the reusability of the framework, because the adaptivity logic in the framework server is separated from the game logic and can be reused for any game, and this separates the computation expenses from the Game Engine. Client is attached to the game engine to observe the players’ actions.

The Framework consists of 5 sub systems presented in Figure 2; namely, Monitoring subsystem, Adaptivity Agent subsystem, Intervention subsystem, Game Model subsystem and Graphical user Interface (GUI) subsystem.

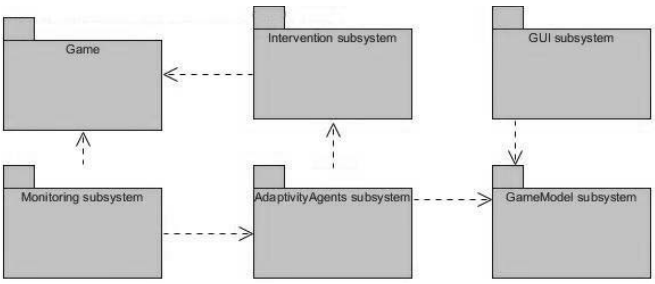


Figure 2: Subsystem Decomposition

The *Monitoring* subsystem contains the modules that are responsible for monitoring the player actions and analyzing those actions to detect events. It is also responsible for logging the player’s actions and events. This subsystem interacts with the game to track what the player is doing. The *Adaptivity Agents* subsystem is responsible for analyzing the data collected by Monitoring subsystem, deciding when the game should be adapted, and sending an adaptivity request to the Intervention subsystem when needed. It is also responsible for updating the game model and updating the skills of the player according to the analyzed events. The *Intervention* subsystem contains modules that adapt the game according to the instructions sent from the Adaptivity subsystem in response to the information collected by Monitoring subsystem. The *GameModel* subsystem provides the representation of the game data as game elements and learning goals. It also contains data about players. The *GUI* subsystem provides the interface to interact with the system. It allows users to add/remove skills from the game model to reflect the required/learned skills and track player skills development.

We used Java Agent DEvelopment Framework (JADE) to implement the agent system. JADE (Bellifemine et al. 2001) is a free cross-platform software developed with Java to implement multi-agent system that follows the guidelines of the Foundation for Intelligent Physical Agents (FIPA). Our Adaptivity agent is implemented by extending JADE agent. jade.core.Agent is the common super class for all user defined agents. To implement a custom agent, we implemented an AdaptivityAgent class that extends jade.core.Agent and implemented setup() for initialization of the agent and takeDown() to implement termination. A class diagram of AdaptivityAgent in Figure 3 shows the relationship with JADE core classes jade.core.Agent and jade.core.behaviours.Behaviour.

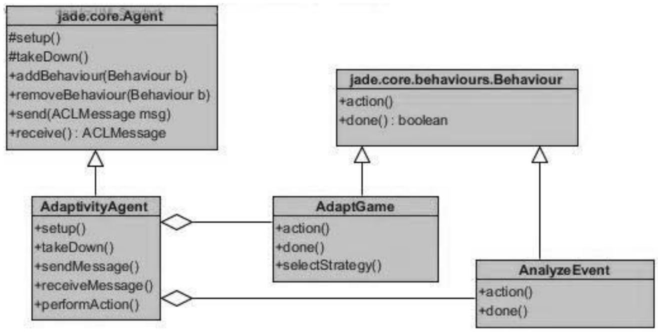


Figure 3: Adaptivity Agent

Actual agent job is accomplished through behaviours. A behaviour is a task to be performed by the agent. The task of the behavior is implemented in Behaviour.action() which is an abstract method. Strategy Pattern is used to allow the running of different set of AI and ML algorithms by the agent. Developers can add new AI algorithms by implementing a new class that implements the AdaptiveStrategy Interface. The strategy to be used can be selected in AdaptGame.action(). Class diagram shown in Figure 4.

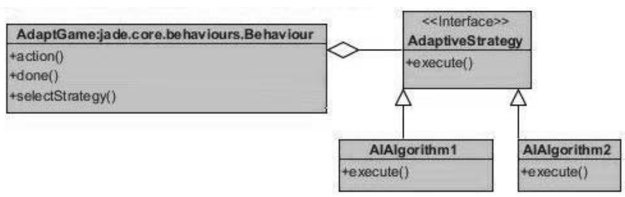


Figure 4: AdaptiveStrategy

The starting point is the AdaptivityServer Class. It is responsible for getting the information that the monitoring subsystem collects through getInformationFromGame() and sending the adaptivity instructions to the game through sendAdaptivityInstructions(). The AdaptivityServer instantiates the AdaptivityAgent and GameModel that includes one or more PlayerDataModel instances representing players who are playing the game. Figure 5 shows class diagram of the Adaptivity Server.

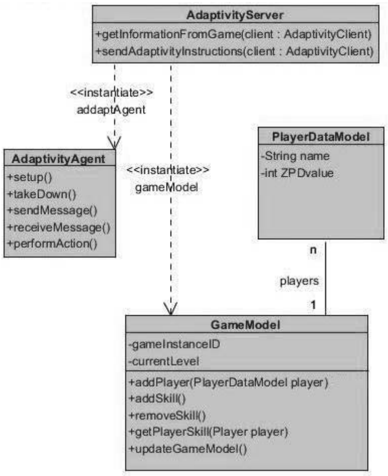


Figure 5: Adaptivity Server

GameModel is the class that keep the data about the game and is illustrated in Figure 6. It has a gameInstanceID to allow the server to distinguish different game instances that are running on the server. It has also information about current difficulty level. The GameModel has list of AdaptiveElements that are the game elements that can be adapted, and a set of LearningGoals that represent learning goals of the game, which can be used by game developer while developing the adaptivity algorithm. GameModel has a set of Skill that are the skills required in the game, details of skills will be disccussed later while discussing PlayerDataModel.

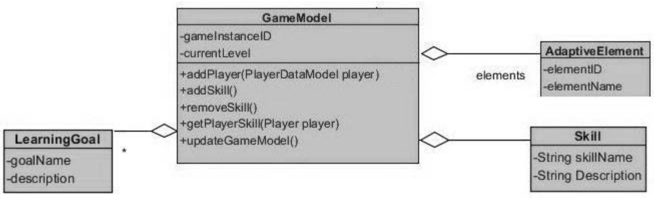


Figure 6: Game Model

Player model class diagram (Figure 7) shows how we model players and their skills. We also store an estimate of where the player is in the ZPD curve, and the dominating learning style that can be sent to AdaptivityAgent to be used while adapting the game. Skills are modeled in the Skill class. Each Skill might require that other skills have a minimum value before a player can start practicing this new skill; this relation is modeled by an adaptation of composite pattern that skill requirement is a skill that has a minRequiredVlaue. PlayerSkill is a Skill with one or more values. Each value is associated with a timestamp to be used to track skill development history. Each player has set of PlayerSkills. We used the precedence diagram like the one presented in Figure 1 to model skills in our framework,

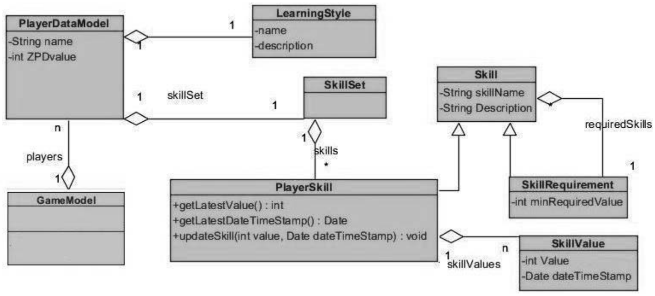


Figure 7: Player Data Model

The aim of Adaptivity Client is to be attached to the game to monitor the player's actions, to send messages to server, and to wait for adaptivity instructions. Because components of this package are highly depend on the game (e.g adaptive game elements and monitoring events), we provided abstract classes that have the functionality of the framework but leave the exact implementation to be implemented by the game developer.

The Graphical User Interface of our framework consists of three views. TutorView (Figure 8-top) that allow tutor to update the game by adding/removing new skills and adding new monitoring techniques. As well as viewing player performance. PlayerView (Figure 8-buttom) is the view where player can see information about his or her skills. And LogView is the view that shows log of player actions that can be checked by teacher and parent for observing player actions.

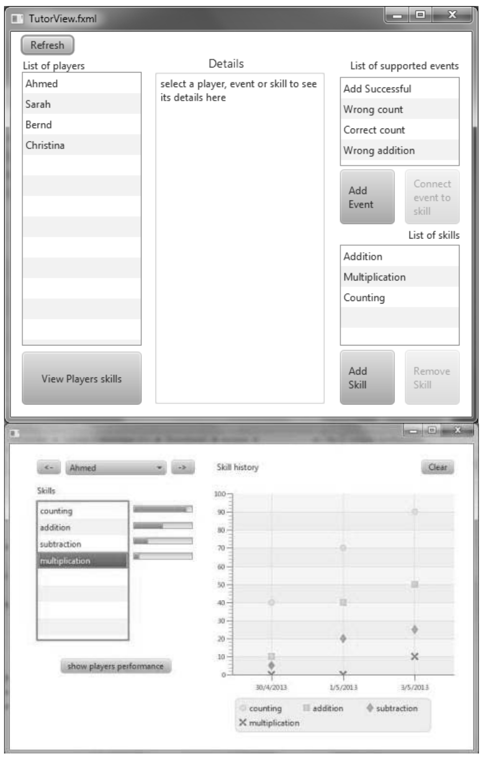


Figure 8: TutorView and PlayerView

EVALUATION

The goal of the evaluation is to check how usable our interface is, and how easy and fast users can perform different functions using the framework. We performed a

Usability test for the different functions supported by our Server. We tracked the time and number of clicks needed for each task and whether the user was able to complete the task successfully or not. Then we asked users to fill out a questionnaire to measure their satisfaction with our application. We chose a random sample of 26 persons. We asked them to perform the following tasks:

- view skill details
- add a new skill
- remove skill
- view event details
- add event
- connect event to skill
- view player skills
- view player details

The questionnaire measures these usability metrics: ease of each task, ease of use of the whole application, suitability for task, degree to which application meets user expectations, and whether the user has full control of the application. We used UsabilityHub (http://www.navflow.com) to perform the test of navigation flow and testing different tasks. Using this website we can upload screenshots of different views of application and ask users to perform tasks. Each task can contain one or more steps, and for each step we can define areas of the screen that should be considered "correct" and if the user clicks on the correct area of the screen, he or she will be navigated to the next step.

From our test we found that the the application is satisfactory for users with regards to usability metrics. In this section we will discuss the results of our test in detail.

Task correctness: In this section we will present the results we get by asking 26 testers to perform some tasks and we perceived their interaction with the application to see whether they perform the task correctly or not in their first try. Table 1 shows the results we got for different functions. We see that at least half of participants intutively perform tasks correctly without single wrong click for the most sophisticated tasks in less than half a minute each task

Table 1: Task Correctness

Task	Num of clicks	Success rate	Avg duration
View skill details	1	54%	16 sec
Add new skill	2	50%	27 sec
Remove skill	2	89%	18 sec
View event details	1	90%	15 sec
Add event	3	70%	31 sec
Connect event to skill	4	54%	24 sec
View player skills	2	54%	21 sec
View player details	1	93.5%	7 sec

Ease of use (Task-based): We asked testers after each individual task to evaluate how easy the task was performed. Choices are from 1 (too difficult) till 5 (too easy).
This survey included 26 participants and results are shown in Table 2

Table 2: Ease of Tasks

Task	Ease of use (mean)	Ease of use (STD)
View skill details	3.615	1.242
Add new skill	4	1.208
Remove skill	4.346	1.036
View event details	3.846	1.350
Add event	4.192	1.075
Connect event to skill	4	0.961
View player skills	3.962	1.018
View player details	4.231	1.049

General application satisfaction: Using a post task questionnaire, after completion of all tasks we asked testers to evaluate their satisfaction of our application with regards to the following aspects: ease of use, suitability to performed tasks, that the application met their expectations, and whether they have the full control on the application. The scale is from 1 (strongly unsatisfying) to 5 (strongly satisfying). From data in Table 3, we can find that the users found our framework easy to use and suitable to the performed tasks

Table 3: User Satisfaction

Measure	Mean	STD
Ease of use	3,846	1,231
Suitability to tasks	3,846	0,988
Meet expectations	3,654	1,207
Control	3,731	1,346

CONCLUSION AND FUTURE WORK

Adaptivity in the context of educational serious games is essential and beneficial for learning motivation and outcome. This is especially true today, when the ability of traditional education to motivate the younger generations appears to have decreased. These generations spend much of their time playing games. Using games for education is expected to raise motivation for learning (Moreno-Ger et al. 2007). Furthermore, adaptive games are a possible solution to enhance current educational techniques, and to introduce interactivity to traditional teaching methods. In this work, we performed a background survey on adaptivity in educational games and related concepts of learning, reviewed the current state of AI and Agent systems, and presented a framework that can be used to develop Adaptivity in educational serious games. Our framework helps teachers, parents and learners themselves to track the development of a learner's skills. It also helps developers to make their games adaptive to the skills of the learner. We believe that using the proposed framework will help developers and teachers/domain experts to make games that are more engaging, enjoyable and motivating for learners. By using this framework learners can enjoy and learn at the same time and learn as effectively as if they had a personal tutor.

Future Work

Adaptivity in educational serious games is important but remains difficult and complex to implement. Thus, many enhancements can be included to the our framework to increase its effectiveness or to add new functionalities. These could include the following:

- The integration of the SchuhPlatter framework, which is used to balance the game manually at real time

(Ismailovic et al. 2012). By integrating SchuhPlatter to our adaptivity framework, we can give teachers and parent opportunity to adapt the game manually and interact with students and children when needed.

- Multiplayer and social interaction: Social games are attracting more players and engage them in social interaction by playing the game. Multiplayer games and social games can motivate players to learn and allowing players to learn from each other. By considering muliplayer and social games in our framework, we can enrich the learner's game experience and learning experience as well.
- Adapting to cultural aspects: It's extremely challenging and important to adapt games to local cultural aspects. Players who come from different cultures behave differently and perceive things differently (Demeestera 1999). By considering adaptivity with regards to culture, both the game and learning experience will be enhanced.

REFERENCES

Bellifemine F.; A. Poggi; and G. Rimassa. 2001 "Developing multi-agent systems with JADE" In *Software – practice & Experience.* John Wiley & Sons, Ltd, vol no. 31,2001, pages 103-128.

Bellotti F.; R. Berta; A. de Gloria; and L. Primavera. 2009 "Adaptive experience engine for serious games" In *IEEE Transactions on Computational Intelligence and AI in games*.

Brisson, A.; G. Pereira; R. Prada, A. Paiva; S. Louchart; N. Suttie; T. Lim; R. Lopes; R. Bidarra; F. Bellotti; M. Kravik; and M. F. Oliveira. 2012 "Artificial intelligence and personalization opportunities for serious games" In *AIIDE Joint Workshop*.

Csikszentmihalyi M. 1990. *Flow: The Psychology of Optimal Experience*. Harper & Row.

Demeestera. M. 1999. "Cultural aspects of information technology implementation". In *International Journal of Medical Informatics*, 56(1-3):25-41, December.

Doignon JP and Falmagne JC. 1985 " Spaces for the assessment of knowledge" In *International journal of man-machine studies* : 23.2:175-196.

Haladjian, J.; D. Ismailović, B. Köhler; and B. Brügge. 2012. A quick prototyping framework for adaptive serious games with 2D physics on mobile touch devices. In *IADIS International Conference Mobile Learning* (197-204).

Harteveld C.; R. Guimarães; I. S. Mayer; and R. Bidarra. 2010. "Balaning play, meaning and reality: The design philosophy of levee patroller" In *Simulation & Gaming,* 3(43):316 - 340

Hunicke R. and V. Chapman. 2004. AI for dynamic difficulty adjustment in games." In *Challenges in Game Artificial Intelligence*. AAAI.

Hocine, N. and A. Gouaich. 2011. "Agent programming and adaptive serious games: A survey of the state of the art." In *10th Int. Conf. on Autonomous Agents and Multiagent Systems*.

Ismailovic, D. 2012. Application of Adaptivity to Serious games. PhD dissertation, Technical University of, Munich.

Ismailovic D.; J. Haladjin; B. Kohler; D. Pagano; and B. Brugge 2012.. Adaptive serious game development. In 2nd *International Workshop on Games and Software Engineering (GAS),* 23-26

Johnson, W.L; H. Vilhjalmsson; and S. Marsella. 2005. "Serious games for language learning: How much game, how much ai?" In *Artifial Intelligence in Education*, pages 306-313.

Kolb, A. Y.; and D. A. Kolb. 2005„"Learning styles and learning spaces: Enhancing experiential learning in higher education". Academy of Management Learning & Education, 4 No. 2:193-212, Jun.

Lopes R. and R. Bidarra. 2011 "Adaptivity challenges in games and simulations: A survey." In *IEEE Transactions on Computational Intelligence and AI in games*, vol 3, no. 2, June.

Manslov j. 2002. "Learning and adaptaion". In Steve Rabin, editor, *AI Game Programming Wisdom*.

Michaud, L. 2010. "Serious games: A 10 billion euro market in 2015" IDATE.

Moreno-Ger, P. P. S. Thomas; I. Martínez-Ortiz; J. L. Sierra; and B. Fernández-Manjón. 2007 „Adaptive units of learning and educational videogames" In *Journal of Interactive Media in Education*

Nieborg, D. B. 2004. "America's army: more than a game?" *Transforming Knowledge into Action through Gaming and Simulation*.

Olaverri Monreal C.; A. Elsherbiny Hasan, and K. Bengler. 2013 "Semi-automatic user stories generation to measure user experience." In *CISTI'2013 - 8Âa Conferência Ibérica de Sistemas e Tecnologias de Informação*.

Peirce N.; O. Conlan; and V. Wade. 2008. "Adaptive educational games: Providing non-invasive personalised learning experiences." In *Second IEEE International Conference on Digital Games and Intelligent Toys Based Education,*

Quinn, C. and L. Neal. 2008. "Serious games for serious topics" In *eLearn 2008,* No.3 (Mar)*: 5.*

Ram, A.; Ontanon S.; and M. Mehta. 2007. "Artificial intelligence for adaptive computer games." In *FLAIRS Conference,* 22-29.

Rieber, L.P. 1996. Seriously considering play: Designing interactive learning environments based on the blending of microworlds, simulations, and games. In *Educational Technology Research and Development*, 44:43-58.

Susi, T.; M. Johannesson; and P. Backlund. 2007 "*Serious game: an overview*" Technical report HS- IKI -TR-07-001 School of Humanities and Informatics, University of Skövde, Sweden (Feb).

Ulicsak, M. 2010. "*Games in education: Serious Games*" A Futurelab Literature Review. FutureLab.

Vygotsky. L.S. 1978 " Mind in Society: Development of Higher Psychological Processes" Harvard university press

Westra, J.; H. van Hasselt; V. Dignum and F. Dignum. 2008. "On-line adapting games using agent organizations.: In *IEEE Symposium on Computational Intelligence and Games* (243-250).

Westra J.; F. Dignum; and V. Dignum. 2011 "Scalable adaptive serious games using agent organizations" In *10th International Conference on Autonomous Agents and Multiagent Systems*, 3:1291-1292,.

WEB REFERENCES

Yates, W. 2007 Tactical Iraqi Language and Culture Training Systems: Lessons Learned from 3rd Battalion http://www.alelo.com/files/Gametech_2011_TILTS_Lessons_Learned_2007.pdf, Page 8

AUTHOR BIOGRAPHY

AHMED ELSHERBINY HASAN was born in Giza, Egypt and recently graduated in Informatics from the Technische Universität München obtaining his M.Sc. degree. In 2007, he obtained a bachelor in Computer Engineering from Cairo University. Between 2007 and 2010 he worked in the development and integration of telecommunication computer systems in Egypt. During his masters at TUM, he worked in research projects at the faculty of Informatics and Institute of Ergonomics in the area of HCI. His research interests are Artificial Intelligence applications and adaptive systems, HMI and applications in an automotive context.
Email: Hasan@in.tum.de

Multi Agent Software Engineering (MaSE) and Agile Methodology for Game Development

Rula Al-Azawi
Gulf College, Muscat, Oman
rula@gulfcollegeoman.com

Aladdin Ayesh
DMU University, Leicester,UK
aayesh@dmu.ac.uk

Khalfan Abdullah AL-Masruri
HCT, Muscat, Oman
khalfan.almasruri@hct.edu.om

Ian Kenny
DMU University, Leicester,UK
ikenny@dmu.ac.uk

Keywords

Agile methodology, Game development methodology, Adaptive development model, Predictive development model, Multi Agent Software Engineering (MaSE), Agile Agent Game Development Methodology (AAGDM)

Abstract

Agile software development methodologies has become very common in game development methodologies. However, such methodologies must be adapted to different game genres. The purpose of this paper is to investigate the existing game development methodologies and provide a new game development methodology suitable for different game genres. Furthermore, the methodology is based on both predictive and adaptive development models.
A critical analysis of Agile game development methodology is presented and we identify the weakness of the current Agile game development. The Agile Agent Game Development Methodology (AAGDM) is introduced as a new hybrid methodology linking the Agent and Agile approach to create a generic game development methodology.

1 Introduction

During the last few decades, several software development methodologies have been used as game development methodologies. Game creation nowadays is an incredibly complex task, much harder than someone might initially imagine. The increased complexity combines with the multidisciplinary nature of the process of game development (art, sound, gameplay, control systems, artificial intelligence, and human factors, among many others). The traditional software development creates a scenario which also increases this complexity. In this connection we need a methodology which takes into account software engineering expertise in the field of games. As we know, the gaming industry is very powerful in the entertainment industry, having billions of dollars in profit and creating trillions of hours of fun [1].
This paper focuses on two archetypical development models, the predictive and the adaptive models [2]. The question here is how we can choose between predictive and the adaptive models in game development methodology and how can components from a variety of game design and development models be integrated into standard development guidelines to create a hybrid methodology which contains the best features from both predictive and adaptive models.
Agent model is an example of predictive development models while Agile model is an example of adaptive development models.
It is important to have a formal understanding of game development process, and how we can create a formal game development methodology that will be generic for different game genres.
The rest of paper is structured as follows: section 2 presents the history of the current game development methodology; section 3 presents a critical analysis of the problems in current game development methodologies; section 4 presents the new game development methodology AAGDM as an approach solving the previous problems; section 5 presents a critical evaluation and discussion of the AAGDM; and section 6 presents a conclusion and future work.

2 History of Game Development Methodologies

Game development deals with large projects employing hundreds of people and development time measured in

years and presents unique challenges owing to the fact that teams comes from multiple disciplines which contribute to games. Until now, some game development companies are still using the waterfall methodology but with modifications. A major issue with the game development industries is that many cbompanies adopt a poor methodology for game creation[3].
Most of the linear manner methodologies such as Waterfull, Incremental, Prototyping and Spiral are classified as predictive even if it contains some iteration but it usually follows sequence phases. Prototyping methodology involves breaking the system into small segments. Furthermore, it involves the user in the process.
The Spiral methodology combines the linear and iterative framework. Spiral development breaks the projects into number of cycles, all of which follow a set of increasingly larger steps.
The majority of methodologies taken and used by game developers are predictive which are comprehensively planning as a separate task prior to actual development. These methodologies are also described as adaptive because multiple iterations and prototypes are used to shape a game. Their design is based on feedback and analysis. [2] Our suggested AAGDM is a hybrid of a predictive model by using AOSE methodology and an adaptive model by using Agile methodology.

2.1 Agile Methodology

Agile methodology is based on implementation over documentation with customer collaboration and has the ability to solve problem and change with agility. Agile methodology is used as alternative to the traditional methods in game development. It keeps essential practices of the traditional methodology with focus on other dimensions of the projects such as collaboration with user in all development stage. Furthermore it depends on the iterative and incremental development with very short iteration that helps to provide custom-made solutions.
The main characteristics of Agile methodologies are: customer cooperation, simplicity, individual, interaction, adaptiveness and being incremental. These characteristics are important to understand an approach to game development based on an Agile methodology. [4].
The Agile methodology as mentioned earlier is an iterative and incremental approach and it achieves the quality and productivity through iterations. Each iterations of sprint phase includes a software development team working through a full software development cycle including planning, requirements analysis, design, coding, unit testing, and acceptance testing as shown in Figure 1 which was adapted from [5] and [6]. The

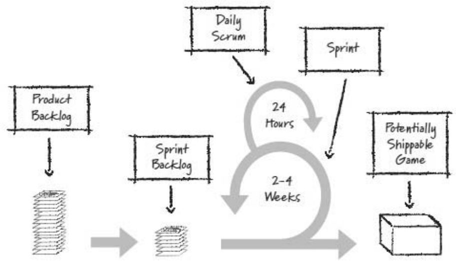


Figure 1: Agile methodology diagram

Agile phase approach diagram which is used by Keith [7] as shown in Figure 2 shows that Agile methodology is based on iterations that could start new iteration before completing the previous iteration. The Agile

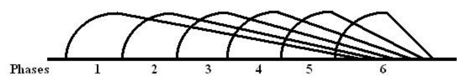


Figure 2: Agile phases approach

methodology has been discussed the use and application through iterative development framework named Scurm [7]. The Scurm game development method is an Agile process which manages game development using iterative and incremental approaches which are the life of the game project. It works in game development methodology by breaking down the process of creating game into series of tasks named "sprints". To facilitate the work with sprints, the game developers break up the game into groups of related tasks or features that must be written in the product backlog. As mentioned in Figure 2, every two to four weeks at the end of sprint phase, the whole teams met to discuss the current state of games to improve version of game to the stakeholders, and to select new tasks from backlog.
According to Keith [7], the Agile game development with Scurm could be labeled as iterative and adaptive model.

2.2 AOSE Methodologies

The relationship between games and AOSE is clear given that software agent or intelligent agents are used as virtual players or actors in many computer games and simulations. The development process is very close to the process of game development[8].
There are several methodologies in AOSE. Each one has its own life cycle. However, some of them are precise only analysis and design such as Gaia, while oth-

ers cover complete life cycle such as Trops, MaSE and Prometheus as shown in Figure 3.
Within the last few years, with the increase in complexity of projects associated with software engineering, many AOSE methodologies have been proposed for development purposes[9]. Nowadays, intelligent agent-based systems are being applied in many domains, including robotics, networking, security, traffic control, games and commerce [10].
The goal when evaluating AOSE methodologies is to discover the most convincing methodology for adaption to game development and incorporation of modifications. Al-Azawi et al [10] focus on comparing different AOSE methodologies from the perspective of the game development domain. The results of their experiment were summarized to select the MaSE as a methodology to be adopted as a game development methodology.
We have selected the MaSE methodology to be adapted for game development methodology for the following reasons:

1. MaSE has a full life cycle. [11].
2. MaSE is influenced by the software engineering root.
3. MaSE is perceived as significant by the agent community [12].
4. MaSE has been selected by [10] as the game development methodology.
5. MaSE has been selected according to many references such as [13] as a methodology for robotics, which is similar to the game area.
6. MaSE has defined the goal at the first stage and each goal has to be associated with its role, which is an important feature of game development.

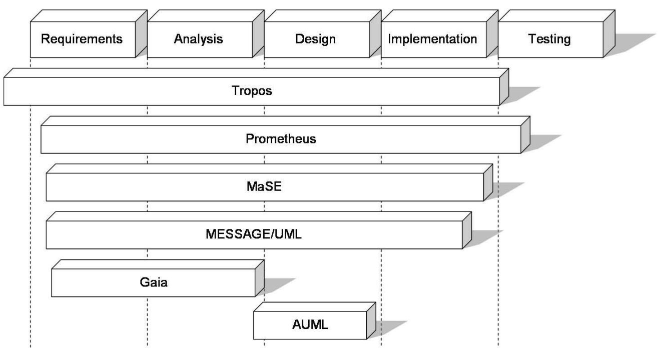


Figure 3: AOSE life cycle coverage

2.2.1 MaSE Methodology

MaSE stands for Multi-Agent Systems Engineering. It is a complete life cycle methodology to help the developer work with a multi-agent system from the start to the end. This means that it describes the process which guides a system developer from an initial system specification to system implementation. In each step, related models are created. Models in one step produce outputs which become inputs to the next step that supports traceability of the models across all of the steps. Furthermore there is possibility for free access between components in each phase.[14]. The goal of MaSE is to guide the system developer from the initial system specification to system implementation[13].

3 Critical Analysis of Current Game Development Methodologies

Games and software engineering have important things they can learn from one another and mostly they share the same methodology and same problems. The game contains a confluence of interesting properties, such as emergence, real time interaction and challenge components that create a new field of study [15]. The software engineering has much to help the game industry to solve problems. The unique aspect of game which is not available in traditional software development is the requirement for game to be 'fun' which has no metric to apply; it is purely subjective.
There are specific features of game development that have been found to prevent the success of great games. The major problems that arise are in the areas of project management. The use of methodology focuses on game development and takes into account the project management concept to help avoid management problems.
After a survey of the current game development methodology problems, we would highlight the main problems found in the literature:

- **Documentation problems:** Lack of documentation is a common source of additional problems. The documentation can be valuable in reducing feature creep. Having a finite amount of documentation is useful when game developers work on difficult projects, as this helps to obtain a good estimate for project scope and schedule [16].
- **Collaboration and Team Management problems:** One of the main problems when creating games is the communication between teams. The teams in games include people with distinct profiles, such as developers, plastic artists, musicians, scriptwriters and designers.

- **Training problems:** One of the biggest problems in Agile game development especially and generally in game development methodology is new employee training.
- **Linear process problems:** Game development is not a linear process [17]. Iteration is the life of game development. Game developers use Waterfall methodology with enhancements, by adding iteration to the methodology.
- **Schedule problems:** According to Flynt et al [17], a key reason for a project being delivered behind schedule is that no target was established. Likewise, problems may occur when a deadline estimate does not include the time needed for communication, lacks documentation or emergent requirements that may alter the system architecture and thereby cause serious problems.
- **Crunch Time problems:** In the game industry, crunch time is a term usually used for the period of work when overload may happen; usually it happens in final weeks before the validation phase or deadline for project delivery.
- **Scope and feature creep problems:** Feature creep is a term used in the game industry when a new functionality is added during the development phase to increase project scope and change schedule time [18].
- **Technology problems:** All games are technology dependent. Technological components generate risks for game projects that can require greater effort and a high investment of time. According to Gershenfeld et al [19],technology risks are generally high when a team works on a new platform because of two risks.

4 Agile-Agent Game Development Methodology(AAGDM)

The suggested methodology AAGDM combines agile methodology that meets the dynamic requirements of the customer with MaSE which is a rapidly development area of research designed to support development of complex and distributed system in open and dynamic environments with the use of intelligent component.
Game development methodology worked better when we used iterative methodology because it allowed to have the features ready soon and to discover and work the fun of the games easier. Ideally, the type of hybrid development methodology approach which we already defined in AAGDM is recommended for use by independent game developers. This possesses a mix of characteristics that would sit somewhere between those of a predictive or adaptive approach to be generic methodology useful for small or large game projects.
Agile methodology is usually used to deal with dynamic changes in requirement specification by the customer, customer involvement in the development phases. For the flexibility in adding new requirements even before game release which does not add extreme cost to the project, Agile game development methodology will be adapted to suggested game development methodology as adaptive model.
AOSE provides such intelligence through agents. Agent may perform the tasks individually. In complex and distributed system, Agents can be used to monitor the interaction among components and to interact as human interaction. The MaSE used in the Sprint phase is at the core of the AAGDM. Each iteration includes analysis, design, implementation, testing and evaluation of MaSE as shown in Figure 4.
The reason to adapt MaSE is that it is the core of Agile, because in complex systems and distributed systems such as games, it is difficult to trace a single point of control, since the objects are distributed [6].
Figure 4 illustrates the suggested game development methodology which we name as Agile-Agent Game Development Methodology (AAGDM).

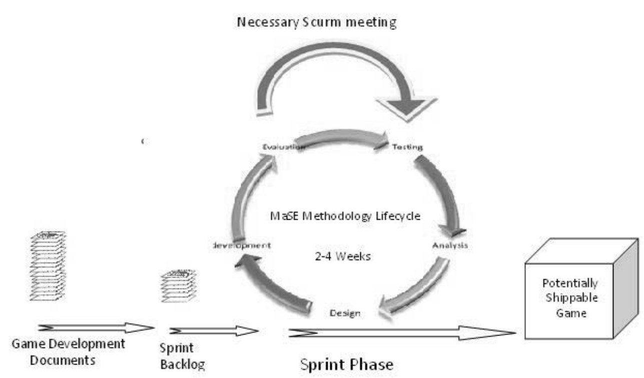


Figure 4: AAGD methodology diagram

5 Critical Analysis of AAGDM

As a first step in game creation, the Game Development Documents (GDD) which is an important step in preproduction phase in the game, being responsible for guiding the project's scope and effect to development and testing phases.
There is no standard way to build GDD, but it must have a comprehensive description of the game in all its aspects. It must also describe the objects and

characters in the game, this effect, how they interact and their role and behavior in the game. The GDD will change many times and will add extra requirements but we should evaluate the risks of changes and if the deadlines can still be met. Later GDD will be translated to a Product Backlog in the production phase, for small games it may be optional, translating the requirements directly as a Product Backlog. This may save time from the team as they go faster to the production, but may also increase risks of feature creep or may not a very entertaining game [4].
If GDD is designed carefully, the project manager can plan the iteration of sprint backlog so that the game is playable at the end of each iteration. This has several benefits. For one, testers can check the game for errors in a playable state which mimics what the end-user would encounter. Having a playable game as early as possible helps the team to see the potential of the end product, and it could be benefit in game publication before final release.
In Figure 5, we noticed that the cost of change of traditional software increased towards the end of the project time, while in Agile it will increase also but normally at the end of the projects [20].
Keith [7] suggests that at the end of sprint, if there is

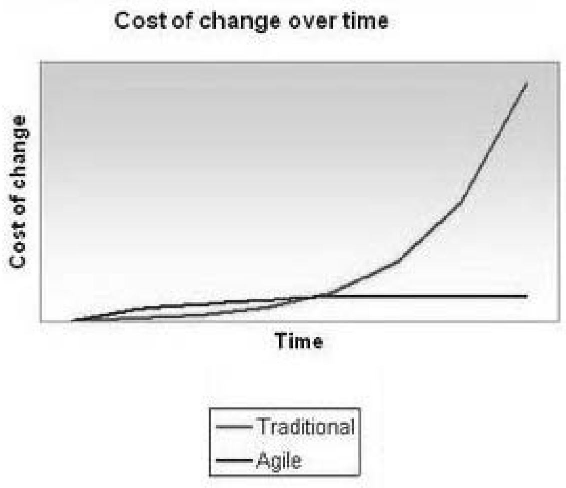


Figure 5: cost of change over time

still work that are still under development. The goal is to achieve a continuous flow in the content of creation as shown in Figure 2 which is the core concept in AAGDM.
The current agile game development faces challenges with distributed, complex and open systems. To overcome this challenge, the component must be capable of automatically interacting with each other and some artificial intelligence must be brought into the components of the game.
AOSE provides such intelligence through agents. Agent may perform the tasks individually. In complex and distributed system, Agents can be used to monitor the interaction among components and to interact as human interaction. MaSE has been adapted to our suggested game development methodology as a predictive model which depends on a linear process and has good game design documents [21].
When AAGDM used Agile concepts, we improved the quality and efficiency of large, complex games projects. Furthermore, it strengthens the communication between the developer and the end user.
The management is important in the game industry. Poor management can negatively affect the best of teams. As the complexity in the game and number of teams increases, good communication in a company is necessary for success. Agile methodology usually depends on daily Scurm meeting to get good communication, but in many cases there is no need to discuss daily because it is only a waste of time for the teams.
In AASDM, we suggested that meeting is not necessary on a daily basis to save time. It should only be done when important issues arise from the multidisciplinary teams such as artists, musicians, developer and clients. Furthermore, the group may have sub-groups such as AI team or a textures team because AAGDM creates functional unit combinations of specialties. An example would be a unit composed of two programmers, a texture artist and an animator. Combining groups does seem to enhance communication across disciplines. Bringing the diverse groups together can enhance understanding and communication between teams. AAGDM is an iterative methodology that focuses on delivery features. The AAGDM has the ability to start dealing with new features before completing a current feature. In this case, game development duration will be reduced because there is no time wasted on waiting.
AAGDM reduces documentation by creating GDD and dividing these into sprint. AAGDM prefer software development over documentation. The game documentation is important and required in the analysis and design phases because we need those details to maintain games or to create new versions of the game. AAGDM is not a linear process, it is an iterative process. Thus, if an interesting feature is discovered, it must be analyzed in terms of its risk and, if viable, it should be added to the project schedule [18].
Keith [7] suggests that at the end of sprint, there is still work that may be under development. The goal is to achieve a continuous flow in the content of creation as shown in Figure 2 which is a core concept in AAGDM.

6 Conclusion and Future Work

Agile and Agent software development approach are the latest trends in software engineering and they found that sometimes a combination of those models may be more suitable for the developers and products. AAGDM is an example of this combination and we have used it in game development.
In this paper, we have introduced new game development methodology AAGDM which combines agile methodology that meets the dynamic requirements of the customer with AOSE which is a rapidly development area of research designed to support development of complex and distributed system in open and dynamic environments with the use of intelligent component.
Game development methodology worked better when we used iterative methodology because it made it possible to have the features ready soon and to discover and work the fun of the games easier.
Our AAGDM solved most of the previous problems in game development by considering being suitable for searcher and professional in the industry.
Future work in this line of research includes evaluating the performance of AAGDM.
Ideally, the type of hybrid development methodology approach which we already defined in AAGDM is recommended for use by independent game developers. This possesses a mix of characteristics that would sit somewhere between those of a predictive or adaptive approach to be generic methodology useful for small or large game projects.

References

[1] F. Petrillo and M. Pimenta, "Is agility out there?: agile practices in game development," *SIGDOC '10 Proceedings of the 28th ACM International Conference on Design of Communication*, pp. 9–15, 2010.

[2] L. Hunt, "Predictive and adaptive game development a practical application of development models to the independent video game industry," Master's thesis, School of Communications and Arts, 2011.

[3] C. Kanode and H. Haddad, "Software Engineering Challenges in Game Development," in *Information Technology New Generations 2009 ITNG 09 Sixth International Conference on*, pp. 260–265, IEEE Computer Society, 2009.

[4] A. Godoy and E. Barbosa, "Game-Scrum: An Approach to Agile Game Development," in *Proceedings of SBGames 2010 Computing* (I. S. F. SC, ed.), pp. 292–295, November 8th-10th 2010.

[5] H. Takeuchi and I. Nonaka, "The new product development game," in *Harvard Business Review*, pp. 137–146., January- February 1986.

[6] M.Nachamai, M.Senthil, and V.Tapaska, "Enacted software development process based on agile and agent," *International Journal of Engineering Science and Technology (IJEST)*, vol. 3, pp. 8019–8029, November 2011.

[7] C. Keith, *Agile game development with Scrum.* Addison-Wesley Signature Series, 2010.

[8] Gomez-Rodriguez, A. and Gonzalez-Moreno, J.C. and Ramos-Valcarcel, D. and Vazquez-Lopez, L, "Modeling serious games using AOSE methodologies," in *11th International Conference on Intelligent Systems Design and Applications (ISDA)*, pp. 53–58, 2011.

[9] O. Akbari, "A survey of agent-oriented software engineering paradigm: Towards its industrial acceptance," *Journal of Computer Engineering Research*, vol. 1, pp. 14–28, 2010.

[10] R. Al-Azawi, A. Ayesh, I. Kenny, and K. AL-Masruria, "Towards an aose: Game development methodology," in *Distributed Computing and Artificial Intelligence, 10th International Conference. Advances in Intelligent and Soft-Computing series of Springer*, May 2013.

[11] J. Sudeikat and L. Braubach, "Evaluation of agent oriented software methodologies examination of the gap between modeling and platform," in *Agent-Oriented Software Engineering, P. Giorgini, J. P. M uller, and J. Odell, Eds. Lecture Notes in Computer Science*, vol. 3382, (Berlin, Germany), pp. 126–141, Springer Verlag, 2005.

[12] Dam, K and Winikoff , M, "Comparing agent-oriented methodologies," in *Proceedings of the Fifth International Bi-Conference Workshop on Agent-Oriented Information Systems (AAMAS).*, vol. 3030, (Melbourn, Australia.), pp. 78–93, Lecture Notes in Computer Science; Springer Berlin / Heidelberg, 2004.

[13] S. DeLoach, E. T. Matson, and Y. Li., "Applying agent oriented software engineering to cooperative robotics," in *the The 15th International FLAIRS Conference (FLAIRS 2002)*, (Pensacola, Florida,), pp. 391–396, May 2002.

[14] S. DeLoach, "Multiagent systems engineering of organization-based multiagent systems," in *SELMAS 05: Proceedings of The 4th International Workshop on Software Engineering for Large-Scale Multi-Agent Systems*, (New York, NY, USA), pp. 1–7, ACM., July 2005.

[15] C. Lewis and J. Whitehead, "The whats and the whys of games and software engineering," in *Proceedings of the 1st International Workshop on Games and Software Engineering, GAS '11*, (New York, New York, USA), pp. 1–4, ACM Press, May 2011.

[16] R. Kortmann and C. Harteveld, "Agile game development: lessons learned from software engineering," in *Proceedings of the 40th conference of the international simulation and gaming association*, 2009.

[17] J. Flynt and O. Salem, *Software Engineering for Game Developers*. Course Technology Ptr, November 2005.

[18] F. Petrillo and M. Pimenta, "Houston, we have a problem...: a survey of actual problems in computer games development," in *In SAC08: Proceedings of the 2008 ACM symposium on Applied computing*, pp. 707–711, ACM, 2008.

[19] M. L. A. Gershenfeld and C. Barajas., *Game plan: the insiders guide to breaking in and succeeding in the computer and video game business*. St. Martins Griffin Press,New York, 2003.

[20] V. Szalvay, "An Introduction to Agile Software Development," *Danube Technologies, Inc., Bellevue, WA*, no. November, 2004.

[21] A. Molesini, O.Andrea, and V. Mirko, "Environment in agent-oriented software engineering methodologies," *Multiagent and Grid Systems - Engineering Environments in Multiagent Systems*, vol. 5, pp. 37–57, January 2009.

Biography

My name is Rula Alazawi, I am working in Gulf college-Muscat- Oman which is affiliated with Staffordshire University-UK as Award leader in computing department. Currently I am PhD student in De-Montfort university- UK. My PhD research area focuses on game development methodology and agent-oriented software engineering methodology. My Past employment was in Amman private university - Jordan

AUTHOR LISTING

AUTHOR LISTING