FUTURE COMPUTING 2010

The Second International Conference on Future Computational Technologies and Applications

November 21-26, 2010 - Lisbon, Portugal

ComputationWorld 2010 Editors

Ali Beklen, IBM Turkey, Turkey
Jorge Ejarque, Barcelona Supercomputing Center, Spain
Wolfgang Gentzsch, EU Project DEISA, Board of Directors of OGF, Germany
Teemu Kanstren, VTT, Finland
Arne Koschel, Fachhochschule Hannover, Germany
Yong Woo Lee, University of Seoul, Korea
Li Li, Avaya Labs Research - Basking Ridge, USA
Michal Zemlicka, Charles University - Prague, Czech Republic

FUTURE COMPUTING 2010

Foreword

The Second International Conference on Future Computational Technologies and Applications [FUTURE COMPUTING 2010] was held between November 21 and 26 in Lisbon, Portugal. The target was to cover (i) the advanced research on computational techniques that apply the newest human-like decisions, and (ii) applications on various domains. The new development led to special computational facets on mechanism-oriented computing, large-scale computing and technology-oriented computing. They are largely expected to play an important role in cloud systems, on-demand services, autonomic systems, and pervasive applications and services.

We take here the opportunity to warmly thank all the members of the FUTURE COMPUTING 2010 Technical Program Committee, as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to FUTURE COMPUTING 2010. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the FUTURE COMPUTING 2010 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that FUTURE COMPUTING 2010 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the area of future computational technologies and applications.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the beautiful surroundings of Lisbon, Portugal.

FUTURE COMPUTING 2010 Chairs:

Marek J. Druzdzel, University of Pittsburgh, USA
Wolfgang Gentzsch, EU DEISA Project / Open Grid Forum, Germany
Franceso Guim, Intel Corporation, Spain
Hiroyuki Sato, The University of Tokyo, Japan
Cristina Seceleanu, Mälardalen University, Sweden
Vladimir Stantchev, Berlin Institute of Technology, Germany
Christian Tchepnda, France Telecom - Orange Group, France

FUTURE COMPUTING 2010

Committee

FUTURE COMPUTING Advisory Chairs

Academy
Vladimir Stantchev, Berlin Institute of Technology, Germany
Marek J. Druzdzel, University of Pittsburgh, USA
Hiroyuki Sato, The University of Tokyo, Japan
Cristina Seceleanu, Mälardalen University, Sweden

Industry
Christian Tchepnda, France Telecom - Orange Group, France
Wolfgang Gentzsch, EU DEISA Project / Open Grid Forum, Germany
Franceso Guim, Intel Corporation, Spain

FUTURE COMPUTING 2010 Technical Program Committee

J. Enrique Agudo, University of Extremadura, Spain
Andreas Albrecht, Queen's University Belfast, UK
Eduardo Alonso, City University London, UK
Angeliki Antoniou, University of Peloponnese - Tripolis, Greece
Eduard Céspedes Borràs, Universitat Autonoma de Barcelona (UAB), Spain
Anisoara Calinescu, University of Oxford, UK
Radu Calinescu, Aston University, UK
Miriam A. M. Capretz, The University of Western Ontario, Canada
Massimiliano Caramia, University of Rome "Tor Vergata" - Rome, Italy
Gang Chen, Lingcore Laboratory, USA
Leandro Dias da Silva, Federal University of Alagoas, Brazil
Carlos Léon de Mora, University of Seville, Spain
Sebastian Dochow, Albert-Ludwigs-Universität Freiburg, Germany
Marek J. Druzdzel, University of Pittsburgh, USA
Atilla Elçi, Middle East Technical University Northern Cyprus, North Cyprus
Francesco Fontanella, Università degli Studi di Cassino, Italy
Ivan Ganchev, University of Limerick, Ireland
Wolfgang Gentzsch, EU DEISA Project / Open Grid Forum, Germany
Victor Govindaswamy, Texas A&M University - Texarkana, USA
Michael Grottke, University of Erlangen-Nuremberg, Germany
Franceso Guim, Intel Corporation, Spain
Abdelhakim Hafid, University of Montreal, Canada
Muhammad Iftikhar, Universiti Malaysia Sabah (UMS), Malaysia
Li Jin, University of Westminster - Harrow, UK
Dalia Kriksciuniene, Vilnius University, Lithuania
Lu Liu, Middlesex University, UK

Jinhua Ma, Hosei University, Japan
Keiji Matsumoto, National Institute of Informatics (NII), Japan
Constandinos Mavromoustakis, University of Nicosia, Cyprus
Natarajan Meghanathan, Jackson State University, USA
Llanos Mora, University of Málaga, Spain
Antonio Muñoz, University of Malaga, Spain
Susana Munoz Hernández, Universidad Politécnica de Madrid, Spain
Kendall E. Nygard, North Dakota State University - Fargo, North Dakota, USA
Mieczyslaw L. Owoc, Wroclaw University of Economics, Poland
Panos M. Pardalos, University of Florida - Gainesville, USA
Ivan Rodero, Rutgers the State University of New Jersey / NSF Center for Autonomic Computing - Piscataway, USA
Aitor Rodriguez, Autonoma University of Barcelona, Spain
Virgilijus Sakalauskas, Vilnius University, Lithuania
Héctor Sánchez, University of Extremadura, Spain
Hiroyuki Sato, The University of Tokyo, Japan
Cristina Seceleanu, Mdh, Sweden
Vladimir Stantchev, Berlin Institute of Technology, Germany
Detlef Streitferdt, ABB Corporate Research - Ladenburg, Germany
Young-Joo Suh, Pohang University of Science and Technology, Korea
Antonio J. Tallón-Ballesteros, University of Seville, Spain
Christian Tchepnda, France Telecom - Orange Group, France
Steffen Thiel, Furtwangen University of Applied Sciences, Germany
Mesbah Uddin, Kyushu University, Japan
Sascha Uhrig, University of Augsburg, Germany
Alexander Wijesinha, Towson University, USA
Zhengping Wu, University of Bridgeport, USA
Hongji Yang, De Montfort University - Leicester, UK
Ludek Žaloudek, Brno University of Technology, Czech Republic
Zhenyu Zhang, Institute of Software / Chinese Academy of Sciences, China

Copyright Information

Table of Contents

Knowledge-based Tool for Software Process Assessment and Improvement

Maria Helynne Lima Silva, Ana Carla de Carvalho Correia, Carlos Eduardo da Silva Costa, Leandro Dias da Silva, Rodrigo de Barros Paes
Universidade Federal de Alagoas
Maceió, Brazil
{mariahelynne, cc.anacarla, neo.edu.costa, leandrodds}@gmail.com, rodrigo@ic.ufal.br

Rodrigo Aiello Praes, João Manoel Silvestre de Sousa, Gustavo Robichez de Carvalho
Pontifícia Universidade Católica do Rio de Janeiro
Rio de Janeiro, Brazil
r.praes@gmail.com, joaomss@primeup.com.br, guga@les.inf.puc-rio.br

***Abstract* - Although many organizations are aware of the importance of using well-defined and organized software development process, they face the problem of how to define and institutionalize it in practice. In order to solve these problems, several process models, maturity models and quality standards have been developed, but the variety of disciplines, methodologies, and best practices is large. This amount of information leads to an overload and can make the task of defining a software process complicated and expensive. To deal with scenario, this paper proposes an approach with two main goals: (i) to develop a model for organizing the knowledge on software engineering; (ii) to develop a software tool to support the model. Once the knowledge base becomes accessible through a tool, organizations can use it as a guide to a software quality improvement program.**

Keywords-Software Quality Improvement; Software Quality Assessment; Knowledge Based Tool.

I. INTRODUCTION

One of the most important factors for the quality of a software product is its development process. A well-defined process helps organizations to follow their schedules, budget and achieve the expected product quality [1]. A standardized process can reduce the room for human mistakes.

Although many organizations are aware about the importance of using well defined and organized software development process, they face the problem of how to define and institutionalize it in the organization.

In order to solve these problems, several process models, maturity models and quality standards have been developed. Typically, these models contain the knowledge acquired by a number of real software development and it is structured through a number of best practices and examples.

The variety of disciplines, methodologies, best practices, is increasing. This amount of information may lead the task of defining a software process to a complicated and expensive problem. Moreover, these models are available in an abstract and scattered way in books, websites, among others. That makes the use of this information even harder for organizations. The Guide to the Software Engineering Body of Knowledge [2] is an example of how this knowledge has been organized. The purpose of the guide is to describe what portion of the Body of Knowledge is generally accepted, to organize that portion, and to provide a topical access to it. The Guide should not be confused with the Body of Knowledge itself, which already exists in the published literature.

To deal with scenario, this paper proposes an approach with two main goals: (i) to develop a model for organizing the knowledge on software engineering, that should allow representing any reference model, such as CMMI-Dev 1.2 (Capability Maturity Model Integration for Development) [3], ISO 15288:2008 (International Organization for Standardization) [4], XP (Extreme Programming) [5] or Scrum practices [6]. (ii) To develop a tool to support the model. The tool should be able to maintain the information through insertion, removal and update, providing a knowledge base of best practices found in literature.

Once the knowledge base becomes accessible through a tool, organizations can use it as a guide to a software quality improvement program. The tool is able to diagnose the riskiest disciplines and provide a complete step-by-step quality improvement plan. The tool is also independent of the evaluation methodology, such as SCAMPI (Standard CMMI Appraisal Method for Process Improvement) [7], or maturity model used as reference, such as CMMI-Dev [3].

This paper is organized as follows. Section II describes the knowledge base model. Section III describes the tool that has been developed. Section IV presents some related work. Section V presents a case study where the tool was actually applied in a simulated scenario. Finally, Section VI briefly discusses the results obtained in the study case and presents the conclusion and future work.

II. THE KNOWLEDGE BASE MODEL

Each element of Figure 1 and the relationships between them are described below. The examples given are for the reference model CMMI-Dev.

CMMI is a process improvement maturity model for the development of products and services. It consists of best practices that address development and maintenance activities that cover the product lifecycle from conception through delivery and maintenance.

CMMI can be used to guide process improvement across a project, a division, or an entire organization. It helps integrate traditionally separate organizational functions, set process improvement goals and priorities, provide guidance for quality processes, and provide a point of reference for appraising current processes [3].

In Figure 1, <Discipline> represents a set of disciplines of Software Engineering, such as Software Requirements, Project Management Software, etc.. Thus, each discipline may be associated with one or more activities in <Activity>. For example, Software Requirements has activities such as Elicit Needs, Non-Functional Requirements, Change Control, among others.

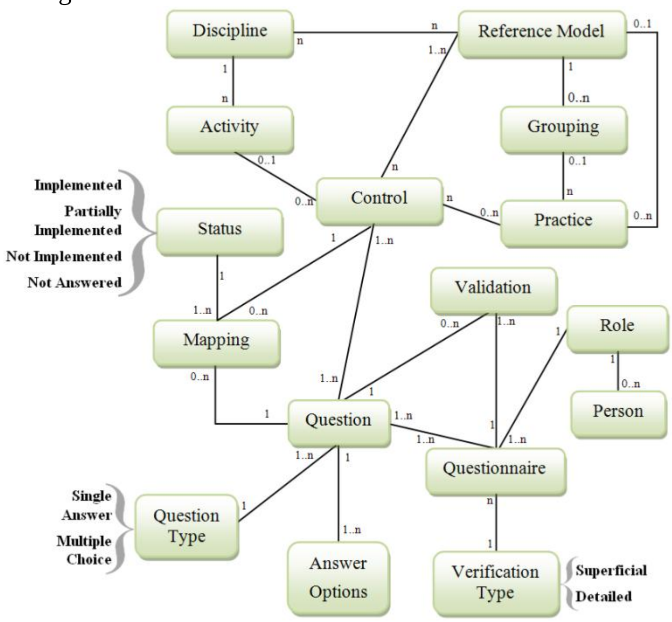


Figure 1. Knowledge base (KB) model

Moreover, <Discipline> has a many-to-many relationship with <Reference Model>, which maintains a set of reference models, for example, CMMI, XP, MPS.BR (Brazilian Software Process Improvement) [8]. Each <Reference Model> describes best practices in software engineering, so it may be related to many <Grouping> or <Practice>.

An example of <Grouping> is Requirements Management (RM). RM is concerned with managing the requirements of the project's product components and identification of inconsistencies between those requirements.

One of the RM's practices is "Requirements Changes Management" since, during the project, requirements may change for a variety of reasons. It is essential to manage these additions and changes efficiently and effectively.

A <Person> has his <Role> in a company. Examples of roles are Manager, Analyst and Developer. The roles can be used to generate specific questionnaires to each company employee.

Thus, a <Questionnaire> is a set of questions filtered according to one or more disciplines, roles and <Verification Type> that determines if the questionnaire will be "Superficial" or "Detailed".

A <Question> may be part of a series of <Questionnaire> and may have several <Answer Options>. A <Question> has a <Question Type> which means a question can be either "Single Answer" or "Multiple Choice".

The <Control> is a checkpoint. It refers to an activity, one or more reference models and one or more practices. The establishment of these relationships makes possible to verify whether the best practices are being applied within the company. The control has a P index (probability), an S index (severity) and an R index (relevance). The product of P*S*R is called PSR index. The general idea of PSR is to indicate quantitatively the risk level if the control is not implemented [9]. The values of P and S should be given by the software engineering expert during the registration of the control. The value of R is determined according to the needs of each company. Possible values and meanings of each index are shown in Table I.

TABLE I
CLASSIFICATION OF PROBABILITY, SEVERITY AND RELEVANCE VALUES

Index Value	PROBABILITY The possibility of the threat causing quality problems	SEVERITY The consequence of the quality problem	RELEVANCE The impairment in the organization	
5	Almost certain ($P \geqslant 95\%$)	Extremely affects quality	Can affect the entire company and the losses are extremely high	Very high
4	Very likely ($65\% \leqslant P < 95\%$)	Very seriously affects quality	Can affect one or more of the company's business and losses are high	High
3	Likely ($35\% \leqslant P < 65\%$)	Seriously affects quality	Can affect a part of the company's business and the losses will be reasonable	Medium
2	Unlikely ($5\% \leqslant P < 35\%$)	Minor affects quality	Can affect a small and specific part of the company's business and the losses will be low	Low
1	Very unlikely ($P < 5\%$)	Hardly affects quality	Can affect a very small and specific part of the company and the losses will be negligible	Very low

Each <Question> should be associated with one or more <Control> in order to investigate the implementation of the practices, activities or disciplines of a reference model.

<Mapping> is a script question and also relates to <Control>. <Mapping> determines what alternatives of each question must be marked so that the <Status> of the <Control> is determined as "Implemented", "Partially Implemented", "Not Implemented" or "Not Answered".

In order to better understand how the controls are used, the Figure 2 describes the use of a control CMMI in tool. It accentuates the elements with descriptions corresponding the CMMI model. In the figure, the <Role> element is instantiated as "Developer". In this way, it is possible to generate a specific questionnaire for this role. The developers will answer the questionnaire and their responses will be analyzed. This questionnaire will contain the question "The clarification of their doubts about the impact that a change can cause is possible because: ". This question is used to verify the application of the control: "Requirements are managed and inconsistencies with project plans and work products are identified". The answers will indicate the level of how implemented the control is, which can assume one of the following values: "Implemented", "Partially Implemented", "Not Implemented" or "Not Answered". As the control is related to the practice "Manage Requirements Changes", it is possible to conclude if it has been applied correctly according to the chosen reference model (CMMI-Dev).

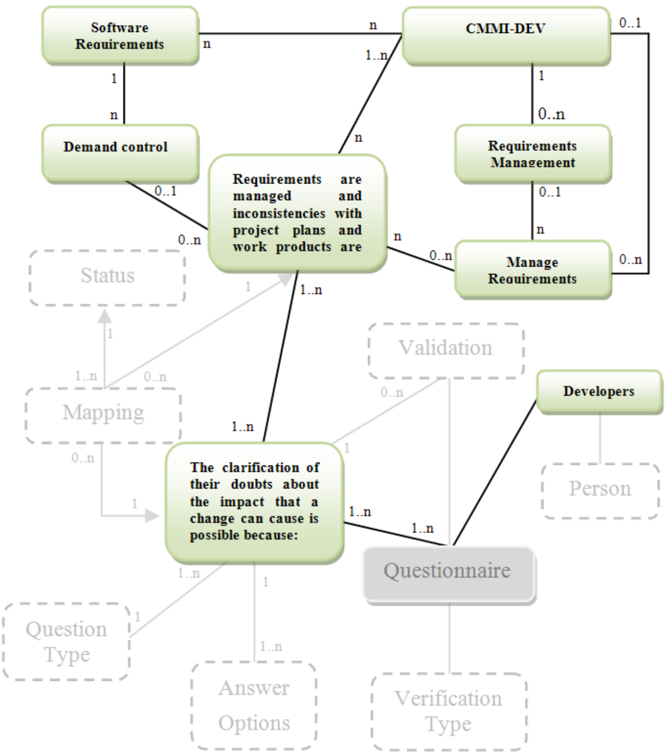


Figure 2. Example of an instantiation of the KB model using a CMMI-Dev control.

III. A SOFTWARE TOOL TO SUPPORT THE MODEL: SÁBIO

Sábio[1] is a web based tool which has been developed in this work. It has two main use cases: (i) maintain a software engineering knowledge base; and (ii) use the knowledge base to evaluate the software process of a given organization [10].

This tool aims to support the model described in Section II. The database contains the entities of the reference model, such as disciplines, practices, activities, roles, questions and controls. The tool is able to generate and send questionnaires to employees of a registered company and, based on the responses, evaluate, in a flexible way, the company's development process from the viewpoint of the employees' roles.

This section shows, step by step, the usage scenario, which is illustrated in Figure 3 and Figure 4. First, we will discuss the usage scenario of the first goal, which consists of maintaining the knowledge base (KB). After that, it will be discussed about how the evaluation process is performed, the second goal.

A. *Maintain a software engineering knowledge base*

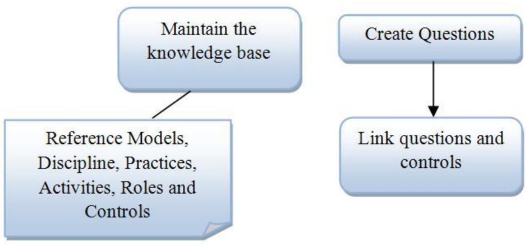


Figure 3. Usage scenario: maintaining the knowledge base.

[1] In Portuguese, the word Sábio is used to designate a wise person.

In order to achieve the first goal, software engineering experts are responsible for two main tasks. One is to build a knowledge base by ensuring quality and integrity. For this purpose, they will register reference models, practices associated with these models and disciplines of software engineering. The tool provides use cases to create, read, update and delete each of these items.

The other task of software engineering experts is to create questions in such a way that the answers extract some information about the development process of the company. When the experts create the question, they have to define its description and answer options. They must associate the question with a discipline and a role, so it is possible to create filters for the generation of the questionnaires. They must also link each question to one or more controls, and for each, write a rule, which will show the status of control.

For example, the expert registers a question and associates it with a control, based on his own knowledge he can create the following rules as scripts:

Question (example): For processes (description of proceedings, techniques, coding standards and templates, etc.) used by you:

a) [ ] There are processes and templates that describe and support the activities that I do. All documentation of these processes is available for use.

b) [ ] There is not a defined process, but we use techniques and practices that support the activities of analysis.

c) [ ] I have means to report my feedback about the activities that I do.

d) [ ] Improvements and changes are implemented in the processes, templates, techniques and practices that support the activities that I do.

e) [ ] Depending on the needs of the project, the templates, the patterns and techniques are adjusted.

f) [ ] I do not have information to answer this question

g) [ ] None of the above.

Control #1 (example) - A program to improve organizational processes should be implemented.

(f OR g) -> Not answered
(c AND d) -> Implemented
(c OR d) -> Partially Implemented
ELSE -> Not Implemented

Control #2 (example) - A useful set of organizational process assets should be established.

(f OR g OR (a AND b)) -> Not answered
(a AND e) -> Implemented
(a OR b OR e) -> Partially implemented
ELSE -> Not Implemented

In this question, two controls are being evaluated, both are related to the activity of Organizational Process Focus (OPF) and they are in accordance with the reference model CMMI-Dev. The status of the control is defined according to the chosen options. In this case, for the control #1, if an employee chooses the options 'f' or 'g', it means that this control was not answered. If the employee selected letter 'c' and 'd', it means that this control has been implemented. If 'c' or 'd' were selected, the control is partially implemented.

In control #2 case, the choices 'f' or 'g', or 'a' and 'b do not answer the question or doesn't have sense, so the control is not answered. The options 'a' and 'e' indicate that the

 ISBN: 978-1-61208-107-6

control is implemented. If ‘a’, ‘b’ or ‘e’ were chosen, the control is partially implemented.

In the case that the chosen answers do not apply in either case, it is considered that the control is not implemented. Sometimes, it is possible to find more than one status for one control, in this case, it is considered the lowest level.

B. *Evaluate the software process*

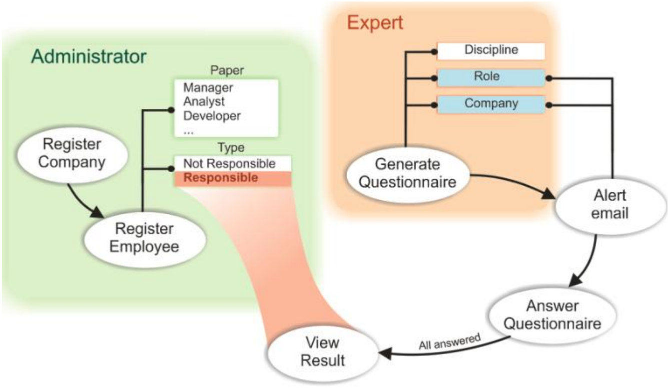


Figure 4. Usage scenario: evaluation process

For the evaluation process of a company, it is necessary to register the company in the database. This is a system administrators’ task.

After the company registration, the employees can be registered. Thus, each employee is linked to a company and to the roles that he performed within the company. The employees are the system users that will be able to answer the questionnaires that are generated by the software engineering experts. An employee may be classified as responsible for the company, this gives him the permission to consult the reports in the end of the assessment.

A questionnaire is a set of questions filtered by disciplines and roles. The software engineering experts generate the questionnaires depending on the needs of the company which has been evaluated. Notice that the questions are already saved in the knowledge base and previously linked to a discipline, roles and several controls.

In the generate questionnaire use case, the expert chooses the disciplines and the role and indicates one or more company to answer the questionnaire. First, the questions are filtered by the disciplines and then by the role. The questions bring the associated controls, for each one, it is assigned the appropriate value of the R index for the company.

All employees of the chosen companies who perform the chosen role are alerted by email that there is a new questionnaire to answer. Once each one log in to the system they will find the questionnaires and they must answer them.

After the answers, the employee clicks on the submit button and each chosen answer is saved in the database. When all the employees of a company answer the questionnaires, the responsible employees are alerted by email that the reporting is now available. Then, the responsible employee logs in to the system and can view the results in a flexible manner, for example, you can generate a report by role, by activity or by the reference model.

The assessment is based on the counting of the controls. Depending on the chosen answers, which are saved in the database, the control status can be determined, thus it is possible count the number of controls according to their status for each answered questions by every employee. In other words, it is made a counting of how many controls are implemented, how many are partially implemented, how many are not implemented and how much were not answered.

Since each control was associated with a discipline, and each questionnaire is directed at a role, it is possible to make this counting in flexibly way. For example, it is possible to analyze the status according to the managers or to have the sum of the controls which are implemented relative to the discipline of Software Requirements.

With the values of P, S and R associated with controls and with the amount of controls for each status, we can determine, using formulas, what activities have the lowest security levels, and the lowest compliance levels. These index values can be calculated in accordance with the equations (1) and (2). So, the activities with the lowest rates must be raised by the report as priorities to improve the development process.

Call:

$$Total_{IC\_PSR} = IC\_PSR + 0{,}5\ PIC\_PSR$$
$$Total_1 = EC\_PSR - NAC\_PSR$$

$$Security\ index = \begin{cases} 0, & if\ Total_1 = 0 \\ \dfrac{Total_{IC\_PSR}}{Total_1}, & if\ Total_1 \neq 0 \end{cases} \quad (1)$$

where:

IC_PSR means the sum of implemented controls’ PSR
PIC_PSR means the sum of partially implemented controls’ PSR
EC_PSR means the sum of evaluated controls’ PSR
NAC_PSR means the sum of not answered controls’ PSR

Call:

$$Total_2 = EC - NAC$$

$$Compliance\ index = \begin{cases} 0, & if\ Total_2 = 0 \\ \dfrac{IC}{Total_2}, & if\ Total_2 \neq 0 \end{cases} \quad (2)$$

where:

EC means the amount of evaluated controls
IC means the amount of implemented controls
NAC means the amount of not answered controls

IV. RELATED WORK

There are some commercial tools with similar goals to the Sábio, such as CMM-Quest [11], Appraisal Wizard [12], and IME Toolkit (Interim Maturity Evaluation Toolkit) [13]. There are also academic tools such as Evaluación Asistida de CMMI-SW (Assisted Evaluation of Capability Maturity Model Integration-Software Engineering) [14] and SPQA.web [15].

CMM-Quest, produced by HM & S IT-Consulting, is a self-appraisal tool for software development organizations to evaluate and analyze their software development processes. The main objective is to support informal assessments based methods (class B and C[2]). The method does not support

[2] The SEI has three classes of methods of evaluation:
- Class A is the most complete, most accurate results, providing a greater understanding of strengths and weaknesses of the organization. The only example of this class is the SCAMPI method.

 ISBN: 978-1-61208-107-6

SCAMPI (Class A). For evaluations, the model supports CMMI-SE/SW/IPPD/SS (Capability Maturity Model Integration - Systems Engineering / Software Engineering / Integrated Product and Process Development / Supplier Sourcing) [16].

Appraisal Wizard, developed by the Integrated System Diagnostics Incorporated is another tool designed to help a team of developers in the assessment of an organization. Considering the work planning, data collection, evaluation team, and generating results for the organization and evaluating these results. It supports practically all models published by SEI (Software Engineering Institute), including CMMI-SE/SW/IPPD/SS model in both representations. It also supports multiple methods of assessment including SCAMPI.

The tools CMM-Quest and Appraisal Wizard work as a repository for collecting information through an assessment. Each piece of information (evidence or opinions of strengths and weaknesses found) are classified and associated with one or more quality standards. Throughout the evaluation process, these tools are used to store data and identify practices that have been implemented.

IME toolkit allows assessments according to the model CMMI-SE/SW. The evaluations include assigning numerical values to the practices. Based on this, the tool generates scores for the process areas. It does not provide support for the SCAMPI assessment method or a detailed evaluation because it is not a tool itself, but a set of Excel spreadsheets.

In [14], the authors propose a tool that provides support for SCAMPI based evaluation. It is possible to register the practice of the CMMI model. The practices are grouped by process areas. The tool is also able to provide compliance reports with both CMMI level 2 and 3.

SPQA.web allows the evaluation of a software development process of an organization. The tool supports the assessment of some process areas of CMMI model and standard ISO/IEC 12207:2002 [17].

Although the related approaches in this section share common goals with this work, there are still some limitations that need to be addressed: (i) Reference model: one of the goals of the proposed approach is to design and build a software engineering knowledge base that is not tied to a particular reference model, as opposite to the most of the related approaches. Then, we provide a higher level model to represent practices that is independent from a particular reference model. The relationship between a generic practice and a reference model is established after the registration of the practice in the base. Furthermore, each practice may be related to more than one reference model. (ii) In the presented related work, there is a consultant that performs the diagnosis. The consultant uses the tools only to register the results of the diagnosis. The proposed work tries to systematize the generation of such diagnosis. In the current version, the way Sábio performs the diagnosis is by using questionnaires and collecting the answers from the stakeholders in order to generate the reports. (iii) The level of details provided in the diagnosis should be configurable. It may be the case that an organization wants to perform a quick and shallow diagnosis. There is also the case that an organization wants to perform a deep and detailed diagnosis. Therefore, the tools must have ways to register in the base both a detailed or superficial practice. Sábio deals with this problem by providing two different levels of practices and consequently two different levels of questionnaires. (iv) When an appraisal is performed, different stakeholders, playing different roles in the development process are involved. Then, the questions that should be asked to each different role should also be different. When a user registers a question in Sábio, he should also select the role that the question should be asked to. (v) Most of the related work provides two outputs: compliant or noncompliant. However, it would be useful to some organizations if the tool provides a report that contains practices that are being followed and the practices that should be followed. Furthermore, even for the same practice, it may have a higher importance in an organizational context than other. For example, an air traffic control company would give a higher relevance to practices related to tests and specification than other company that produces payroll software.

- Class B: A method on a smaller scale, also called mini-appraisal. It goes into less detail than in the class A, and requires less effort.
- Class C: is the least intensive of the three, also called micro-appraisal. Gives some simple idea for the practices employed in an organization.

The only method endorsed formally by the SEI is SCAMPI (Class A). Methods B and C do not have a formal specification by the SEI, leaving its implementation by the concerned.

V. CASE STUDY

To demonstrate the applicability of the tool that has been presented, it was proposed a case study in which three employees of a company were submitted to use the tool. They answered questionnaires for the assessment of the Software Requirements discipline in the development process within the company.

The team was composed of two analysts and one manager, which will be referenced by the names Analyst#1, Analyst#2 and Manager#1.

The case study was divided into four steps:

Step 1 (Company and Employees Registration): In this step, the system administrator registers the company and the employees that will answer the questionnaires.

Step 2 (Questionnaire Preparation): At this stage, the tool was used for generating questionnaires. In this case study, a questionnaire was created to evaluate only the activities of the discipline of Software Requirements.

Step 3 (Assessment and Information Collection): In this step, the team members assessed answered the questionnaire.

Step 4 (Reports Generation): The reports are generated based on the chosen answers and on the information of each evaluated control of the questions. With this information reports were generated showing the status of controls and the activities of Requirements Software.

A. Questionnaire Preparation

Figure 5 shows a screenshot of generation of a Software Requirements Questionnaire.

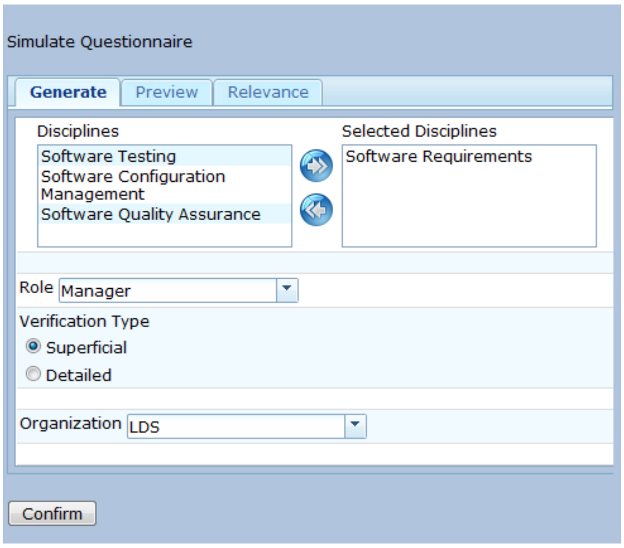


Figure 5. Screenshot, Generating Questionnaire of Software Requirements.

On the "Generate Questionnaire" use case, we generated two detailed questionnaires, one for the manager and one for the analysts. So, the following options were selected:

- Selected Disciplines: Software Requirements;
- Selected Role: Manager and after Analyst;
- Verification type: Detailed.
- Selected Company: The team's company was selected.

Clicking on the "Confirm" button, the system chooses the questions according to what was selected. The way it does this is filtering the questions by the disciplines that have been selected, and then the questions are filtered by the professionals' roles. Finally, the questions will be filtered by their verification type (superficial or detailed).

The expert can view the questions that will be part of the questionnaire on the "Preview" tab and add the relevance of the questions associated controls, tab "Relevance".

Note that the questions, activities and controls were previously registered in the knowledge base by experts in the field.

After that, the system looks for the employees who are already registered in the database and linked to the selected company. In this case, it has found the Analyst#1 and the Analyst#2, when the generated questionnaire was for Analysts, and the Manager#1 when the generated questionnaire was for manager.

Then the system saves the questionnaires in the database. These questionnaires will be answered later by employees when they access the tool. The employees are alerted by a notification email.

B. *Assessment and Information Collection*

When the company's employees access the tool by system login use case, they will be allowed to view the knowledge base and answer their specific questionnaires.

In this case study, the employees accessed the tool and answered 23 detailed questions, taking approximately the following times:

Analyst#1: 2 h 30 min
Analyst#2: 1 h 30 min
Manager#1: 1 h 45 min

At the end of the answers, they just clicked on "Submit" button and all the chosen answers were stored in the database.

When all the company's employees answer their questionnaires, the phase of Reports Generation begins.

C. *Reports Generation*

After all members of the development team answered the questionnaires, the responsible employee was alerted by email that the report generation was available. Thus, it is just log in the system and view the report.

The controls were counted according to their status and separate by disciplines and roles. For an overview, including results from both roles, the values of the results for analysts and managers are added. See the following tables Table II and Table III.

TABLE II
GENERAL RESUME CONTROL BY ACTIVITIES

Activity	Quantity of Controls				Indicator
	Imple-mented	Partially Imple-mented	Not Imple-mented	Total	Compliance Index
Demand Control	6	8	1	15	53,45%
Scope Definition	8	4	0	12	76,96%
Requirements Detailing	10	4	1	15	75,32%
Elicit Needs	4	2	1	7	71,62%
Requirements Management	3	3	3	9	33,33%
Change Control	4	5	3	12	41,95%
Requirements Review	3	2	4	9	50,62%
Requirements Approval	3	2	3	7	57,84%
Non-functional Requirements	1	1	7	9	17,78%
Maintenance and Evolution	7	7	1	15	60,61%
Total	49	38	23	110	57,47%

TABLE III
GENERAL RESUME - CONTROL

Control Status	Quantity
Implemented	49
Partially Implemented	38
Not Implemented	23
Not Answered	0
Total	110

For a better graphical representation, see the following figures Figure 6 and Figure 7.

The Figure 6 displays a graph (Pizza) of the total of evaluated controls, separating them by status.

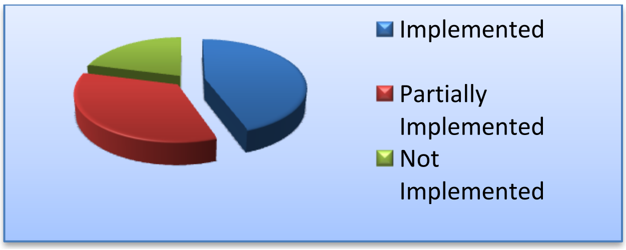


Figure 6. Controls' Status.

The Figure 7 shows a comparison between the activities, ordering by the worst compliance index.

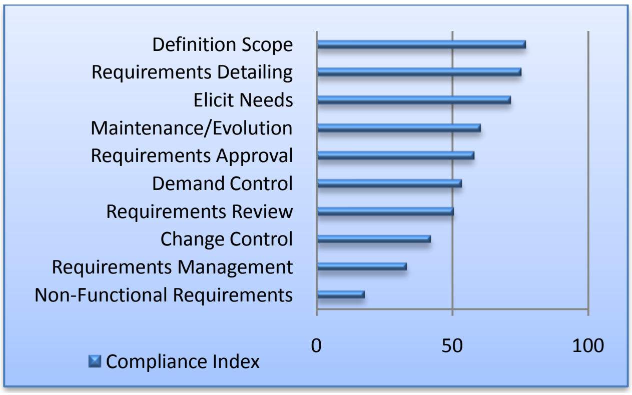


Figure 7. Activities Ordered by Compliance Index.

VI. CONCLUSION AND FUTURE WORK

In the case study, it was found that the activities Non-Functional Requirements and Requirements Management have the lowest compliance index, and therefore they need a plan for improvement.

The main objective of the case study was to apply the tool in a real situation and verify its applicability. Thereby, the knowledge base was consistent and it was possible generate and evaluate the questionnaires.

For the generation of assessments in others disciplines or even in other levels of detail, it would be necessary to use Sábio to register the new knowledge and then generate new questionnaires automatically.

Then, with the tool, it is already possible to create, view, update and delete reference models, practices associated with these models, disciplines of software engineering, roles within the organization, and verification controls. It is also possible to relate questions to controls and generate questionnaires, as well as, interpret the answers chosen and indentify the riskiest disciplines, generating recommendations to improve the software process development.

Other use cases that are also implemented includes the "registration of the organization and user", "user login system", "roles and permissions for each kind of users in the system", and the "collection of the answers" from questionnaires that are being stored in database for evaluation.

With this work, we expected to contribute to the implementation of more efficient development processes, within quality standards. To improve software development, the field of Software Engineering is joining efforts to get better specifications, development and maintenance of systems, applying technologies, practices of project management and other disciplines. Moreover, all this demand reflects a side effect regarding the cost needed to be invested. This is because the development process requires the experience of the various methodologies of Software Engineering and also high qualified professionals.

This research, therefore, created a system that brings together in one environment the knowledge provided by various experts. A first motivation for developing the system was directly related to the possibility of organizing technical knowledge experienced through the use of customizable questionnaires. Another aspect that deserves attention is the aid provided by the tool, through reports, for emphasizing aspects of development that should be improved or met and the guides that explain how the improvements can be performed.

ACKNOWLEDGMENT

This work is partially supported by National Counsel of Technological and Scientific Development - Brazil (CNPq) under the project PROCOCO (620063/2008-4).

REFERENCES

[1] M. B. Chrissis, M. Konrad, and S. Shrum. “CMMI: Guidelines for Process Integration and Product Improvement”. Boston: Addison-Wesley, 2003.
[2] A. Abran, J.W. Moore, P. Bourque, and R. Dupuis, “Guide to the Software Engineering Body of Knowledge (2004 Version)”. IEEE Computer Society Press, Los Alamitos, CA, 2004.
[3] Software Engineering Institute (SEI). “CMMI for Development, Version 1.2”. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2006b.
[4] ____. “ISO/IEC 15288:2008”. Available on-line at http://www.iso.org/iso/catalogue_detail?csnumber=43564/>. Accessed: 06/2010.
[5] K. Beck. “Extreme Programming Explained: Embrace Change”. Addison-Wesley, 1999.
[6] K. Schwaber, and M. Beedle “Agile software development with Scrum”, Prentice-Hall, 2002.
[7] D. M. Ahern, J. Armstrong, A. Clouse, J. R. Ferguson, W. Hayes, and K. E. Nidiffer. “CMMI SCAMPI Distilled: Appraisals for Process Improvement”. Addison-Wesley, 2005.
[8] SOFTEX. “MPS.BR - Melhoria de Processo do Software Brasileiro. Guia Geral”. Versão 1.1, 2006b.
[9] R. S. L. Espinha. “Uma Abordagem para a Avaliação de Processos de Desenvolvimento de Software Baseada em Risco e Conformidade”. 2007. Master's thesis - Pontifical Catholic University of Rio de Janeiro, Computer Science Department, Rio de Janeiro, 2007.
[10] R. A. Praes. “Ferramenta para Organização do Conhecimento e Avaliação de Práticas de Engenharia de Software (Sábio)”. 2009. Course Conclusion (Undergraduate) - Pontifical Catholic University of Rio de Janeiro, Computer Science Department, Rio de Janeiro, 2009.
[11] HS&M. CMM Quest v1.2. Available online at <http://www.cmm-quest.com/english/>. Accessed: 06/2010.
[12] ISD – Integrated System Diagnostics Incorporated. Displays Appraisal Wizard Tools Wizard Lite and to evaluate processes. Available online at <http://www.isd-inc.com/tools.appraisalWizard/>. Accessed: 06/2010.
[13] IME Toolkit, 2003. Interim Maturity Evaluation Toolkit, Management Information Systems. Available online at <http://www.man-infosystems. com/IMEtoolkit.htm/>. Accessed: 06/2010.
[14] M. Peralta, E. Diez, P. Britos, and R. G. Martinez. “Evaluación asistida de CMMI-SW”. Jornadas en Ingenieria de Sistemas Informaticos y de Computacion (JISIC 2004), 2004.
[15] F. J. Pino, F. Garcia, M. Piattini. “A support tool for rapid software process assessment”. Latin America Transactions, IEEE (Revista IEEE América Latina), vol. 5, 2007, p. 218-223.
[16] CMMI Models. Available on-line at http://www.sei.cmu.edu/cmmi/models/models.html/>. Accessed: 06/2010.
[17] ISO/IEC 12207:2002. Information technology - Software life cycle processes. International Organization for Standardization. Geneva. 2002.

Implementing a VoIP SIP Server and User Agent on a Bare PC

Andre L. Alexander, Alexander L. Wijesinha, and Ramesh K. Karne
Department of Computer & Information Sciences
Towson University
Towson, MD 21252
USA

***Abstract*—Bare PC applications run on ordinary desktops and laptops without the support of an operating system or kernel. They provide immunity against attacks targeting an underlying operating system, and have been shown to perform better than applications running on conventional systems due to their reduced overhead. We describe a bare PC SIP server and a SIP user agent designed for VoIP and give details of their internal implementation. The server and user agent include streamlined SIP functions and message handling, efficient CPU tasking, protocol and application intertwining, and direct Ethernet-level data manipulation. The SIP server provides registration, proxy, and redirection services, and the user agent is integrated with lean implementations of the necessary protocols within the bare PC softphone. Bare PC SIP servers and SIP softphones can be used for building secure and efficient low-cost VoIP systems, or deployed in existing VoIP networks with conventional SIP servers and user agents.**

Keywords-bare machine computing; bare PC; SIP implementation; SIP server; SIP user agent; VoIP.

I. INTRODUCTION

Telephony systems over the Internet continue to evolve with the development of new or enhanced VoIP technologies. SIP [1] is an important protocol that provides support for VoIP by handling functions such as call set up, user authentication, user registration and location, and billing support. Although SIP is a general-purpose protocol that can also be used for video conferencing, instant messaging and gaming, it is predominantly used today in VoIP systems. Conventional SIP implementations in servers and softphones require the support of a traditional operating system (OS) such as Windows or Linux, or an OS kernel. SIP phones are also frequently implemented in hardware/firmware typically with an embedded OS. The SIP implementations in OS-based systems take advantage of their rich supporting environment and capabilities and are convenient to use.

However, an OS-based full SIP implementation is not always needed. If a higher level of security or performance is desired at low cost, a customized SIP server or a SIP softphone running on a bare PC (ordinary desktop or laptop without a conventional OS or kernel) would be more easily secured or designed for high performance. For example, an OS-based system may be difficult to secure against attacks that target vulnerabilities of the underlying OS. Bare PC systems are inherently immune against such attacks since they have no OS. In addition, compared to their OS-based counterparts, they also have reduced code complexity and code size, making it easier to analyze their code for security flaws. Moreover, due to their simplicity and the limited services they offer, they also have fewer avenues open for attackers to exploit. Also, studies of bare PC Web servers [2] and email servers [3] have shown that they perform better than their OS counterparts.

Thus, a SIP server or SIP user agent running on a bare PC can be expected to provide secure and efficient low-cost operation. Moreover, since there is no OS, lean versions of the necessary protocols can be intertwined with the bare PC SIP server or SIP softphone application to reduce the overhead of inter-layer communication and improve performance. A preliminary performance study of the bare PC SIP server (see the related work section below) confirms that it performs better than Linux or Windows-based SIP servers with very few exceptions.

In this paper, we describe the design and implementation of a bare PC SIP server and a bare PC SIP UA. In particular, we discuss the details of SIP operations, message handling, and task structure on the bare PC SIP systems. We also examine possible causes for the few performance bottlenecks identified in the bare PC SIP server performance study and note possible future design improvements.

As with other bare PC applications, the SIP server or user agent implementation and interfaces to the hardware constitute a single self-contained executable. The SIP UA is also integrated with the bare PC softphone. The bare PC SIP server and SIP UA implement only the key elements of SIP and have minimal functionality compared to conventional OS-based SIP servers and SIP UAs. Also, the SIP implementations are UDP-based, and the server is stateless. A SIP server implementation over TCP is under development. The bare PC SIP server and bare PC SIP softphone currently run on an IA32 (Intel Architecture 32-bit) or Intel 64-bit architecture in 32-bit mode. They can be used for building secure or high-performance bare PC-only VoIP networks, or interoperate with conventional OS-based SIP servers and SIP softphones as discussed below in the section on testing.

The rest of this paper is organized as follows. In Section II, we briefly survey related work. In Section III, we give an overview of bare machine computing. In Sections IV and V respectively, we describe the design and implementation of the bare PC SIP server and UA. In Section VI, we discuss testing of the SIP server and UAs. In Section VII, we present the conclusion.

II. RELATED WORK

There are numerous implementations of conventional SIP servers and SIP softphones on various OS platforms. These SIP servers and UAs run on conventional OSs. In [4], a SIP server is implemented on top of an existing SIP stack. In [5], SIP servers and SIP UAs are implemented on the Solaris 8 OS. A client-side SIP service offered to all applications based on a low-level SIP API is described in [6]. In [7], the features of a new language StratoSIP for programming UAs that can act respectively as a UA server to one endpoint and as a UA client to another are presented. In [8], the UA is a SIP-based collaborative tool implemented by using existing SIP and SDP stacks. In [9], a Java-based SIP UA is proposed for monitoring manufacturing systems over the Internet. The focus of [10] is a SIP adaptor for both traditional SIP telephony and user lookup on a P2P network that does not have a SIP server. The goal of such SIP servers and SIP UAs is to offer enhanced services to clients by using existing low-level SIP stacks that rely on an OS. In contrast, bare PC SIP servers and UAs that are implemented directly on the hardware will have less overhead and are more suited for secure low-cost environments.

Intertwining bare PC Web server or email server application and application protocols (i.e., HTTP or SMTP) with the TCP protocol contributes to its improved performance over OS-based servers [2, 3]. In [11], the performance of a bare PC SIP server is compared with that of OS-based servers, and it is shown that the bare PC server performs better except in a few cases. It is likely that the performance drops are due to using a simple (non-optimized) search algorithm for user lookup as discussed in the section on server implementation below. The performance study did not discuss the SIP server design details or its implementation. The design, implementation, and performance of a bare PC softphone are discussed in [12, 13]. However, the softphone does not include a SIP UA and hence lacks the ability to communicate with SIP servers and set up calls with other SIP softphones.

III. BARE MACHINE COMPUTING

Bare PC application development is based on the bare machine computing paradigm, also referred to as the dispersed operating system (DOSC) paradigm [14]. In this paradigm, a single self-supporting application object (AO) encapsulating all of the necessary functionality for a few (typically one or two) applications executes on the hardware without an OS. Bare machine applications only use real memory; a hard disk is not used. The AO, which is loaded from a USB flash drive or other portable storage medium, includes the application and boot code. The application code is intertwined with lean implementations of the necessary network and security protocols. If required by the application, the AO also includes cryptographic algorithms, as well as network interface and other device drivers, such as an audio driver in case of the bare PC softphone. The interfaces enabling the application to communicate with the hardware [15] are also included in the AO. The AO code is written in C++ with the exception of some low-level assembler code. The AO itself manages the resources in a bare machine including the CPU and memory. For example, every bare PC AO has a main task that runs whenever no other task is running, and network applications require a Receive (Rcv) task that handles incoming packets. Additional tasks may be used depending on the applications included in the AO, such as an audio task for the bare PC softphone.

IV. BARE PC SIP SERVER IMPLEMENTATION

The bare PC SIP server supports registrar, redirector, or proxy modes with or without authentication. The server is designed in a modular fashion to allow for easy updates and implementation of new features, and to facilitate analysis of the server code. Since the bare PC SIP server implementation is lean, only specific content from an incoming SIP packet is parsed. The bare PC SIP server AO contains about 2000 lines of code.

A. Boot Sequence

The bare PC SIP server is booted by directly loading its AO from a USB flash drive. The protocol/task relationships for the server are shown in Fig. 1. The bare PC SIP Server boot sequence begins when the Main task invokes the DHCP handler to send a DHCP request for an IP address (unless the server has been preconfigured to use a specific IP address). When a response arrives, the Rcv task is invoked to process it. Next, a file containing username and password combinations of authorized users is transferred from another host on the network using an adaptation of trivial FTP. As discussed later, multiple data structures to facilitate server operations such as user lookup, username and password lookup, and state lookup are then created in memory. The last step in the boot process is to display the user interface for administering the server.

B. SIP Server Internals

The bare PC SIP server uses only two CPU tasks, Main and Receive (Rcv), which simplifies task management and increases efficiency. The Main task runs continually and activates the Rcv task whenever packets arrive in the Ethernet buffer and need to be processed. After a response is sent, the Rcv task terminates and the Main task runs again. For example, when the SIP Server AO's Rcv task is activated by the Main task upon the arrival of a SIP request in the Ethernet buffer, a single thread of execution handles the request all the way from the Ethernet level to the SIP (application) level till a response is sent, which simplifies server design and reduces the processing overhead. Thus, if an arriving packet is designated for the default SIP UDP port 5060, the Rcv task causes the Ethernet, IP, and UDP handlers to be invoked to process the respective protocol headers using a single copy of the message. As shown in Fig. 1, the Rcv task only terminates after the SIP request is processed and a SIP response is sent by the server (after invoking the respective protocol handlers to attach the headers).

The bare PC SIP server AO consists of several objects. In addition to the Ethernet, IP, UDP, and SIP objects, the server also requires the DHCP, FTP, and MD5 objects. The role of the DHCP and FTP objects were discussed earlier. The MD5 object is used to provide support for user authentication via standard SIP authentication (i.e., HTTP-Authentication) if it is needed.

ISBN: 978-1-61208-107-6

C. User Database Lookup

After the usernames and passwords from the file are read into memory, the bare PC SIP server runs the sipservergetdb() function to store them in the USER_DATABASE structure:

```
Struct USER_DATABASE {
char username [20];
int username_size;
int username_hash;
char Password [20];
int Password_size;
};
```

The data structures HASH_TABLE and SORTED_TABLE shown below are also used.

```
Struct HASH_TABLE {
int hash_hit;
int hash_reg_db_loc[HASH_REG_DB_SIZE];
int hash_hit_size
};
Struct SORTED_TABLE {
int hash;
int hash_link;
};
```

In essence, the hash of each username is then used as an index into HASH_TABLE, which is used together with SORTED_TABLE to facilitate looking up the user in the USER_DATABASE structure, and retrieving information when making or receiving calls, or registering a user. The HASH_TABLE structure links back to the SORTED_TABLE and USER_DATABASE structures. The details are as follows. First, the hash values are stored in a SORTED_TABLE array (which allows for efficient searching for a given hash value), and each position in the sorted array is linked to the specific HASH_TABLE array corresponding to that hash value. In turn, each position in the HASH_TABLE array corresponds to a user that hashed to that value and contains a link back to the USER_DATABASE entry for that user. The HASH_TABLE structure links the index in the USER_DATABASE structure to the hash value of the SORTED_TABLE as shown in Fig. 2.

The user lookup process in Fig. 3 is done by using two functions: the find_hash_hit() function, which is based on a particular hash value, and the find_user() function that is based on the username and size. In performance tests, this search operation was found to be a likely bottleneck because of the username comparisons triggered by collisions on a single hash value. The find_user() function takes a username and username size as input. It then hashes the username and passes the value to the find_hash_hit() function, which finds the corresponding hash table containing all the users with that same hash value. The hash table is passed back to the find_user() function, which calls the lookup_user() function. The latter goes through each user in that specific hash table and first compares the sizes of the usernames; if they match, it looks for a second match on the full username. If the user is found, the location containing the user's information in the database, including the IP Address and port, is returned. To improve performance, future bare PC SIP server implementations will use adaptations of data structures and search techniques used by popular Linux SIP servers.

D. SIP Message Processing

The siphandler() function manages the processing of received SIP messages. This function, which is called directly by the udp_handler() function after verifying the SIP port in the UDP header, is the key element in the bare PC SIP server. The siphandler() function calls the parse_headers() function which goes through the SIP packet and parses out specific identifiers to identify the type of message (for example, REGISTER, INVITE, ACK, BYE, 180 Ringing, 200 OK and 100 Trying). Within the parse_headers() function are specific functions built to handle the following SIP tags: Header, Via, From, To, Expires, Authorization, Proxy Authorization, CallId, CSeq, Contact, and Content Length. In keeping with the lean SIP implementation, only the indicated tags are parsed to expedite the processing of SIP packets (other tags are bypassed). Once the tags are parsed and the relevant data from the packet is stored, control returns to the siphandler() function. Further processing is determined according to the request_type returned. Only the following SIP messages are routable by the Bare PC SIP Server: Register Invite, 100 Trying, 180 Ringing, 200 OK, Ack, Bye, and Unsupported. When the system (the siphandler function) has decided what to do with the SIP request, processing is carried out to forward the SIP message, or a reply is sent to the SIP User Agent by utilizing the generate_sip_response() function. This function generates the SIP reply (or 100 Trying response) based on the values retrieved earlier by parsing the SIP request. It then calls the sipsenddata() function, which calls the relevant protocol handlers to format the headers in the SIP reply.

Register Message: To process a Register message, the bare PC SIP server parses the Via (IP address:port), From and To (usernames@domain/IP), and Contact tags. It then calls the function check_registered_users(). A process similar to that described earlier is used to determine if the user is already registered (i.e., is found in the Registered_Users_Database). If so, only the relevant information is updated; otherwise, the system stores all necessary information parsed from the SIP request including the username, IP address and port number. This information is used to generate replies back to the UA on future requests until the UA re-registers or one of the parameters is updated. After the information is stored or updated, the server generates a 200 OK message and sends the reply back to the SIP UA.

Invite Message: For an Invite message, the bare PC SIP server parses almost all of the same fields as for the Register message. The server then sends messages to the caller and callee. A 100 Trying message is sent back to the caller letting the UA know that the SIP Server is processing the request. To send this message, the server looks up the IP address of the caller using the process described earlier. It also looks up the registration information for the callee and forwards the Invite message to its UA.

 ISBN: 978-1-61208-107-6

SIP Authentication: The Message format for an Invite request with authentication is shown in Fig. 4. SIP authentication is done by challenging the initial request (Invite or Register) sent by the SIP UA. SIP uses HTTP authentication techniques. The bare PC SIP Server is designed so that each request is not authorized unless it receives the proper response for a given challenge. The server can be configured at start-up to operate with or without authentication. An authorization flag indicates if a particular request is approved or denied based on authentication. The bare PC SIP server processes the initial request, and then sends a challenge response back to the requesting SIP UA. The SIP server generates a challenge response that depends on the values of realm and nonce. The realm is typically set to the domain of the SIP server (for example, barepc.towson.edu or the IP address). The nonce is a string that is randomly generated by the server. Once the server receives the reply to the challenge, the fields in the authorization request are parsed from the SIP packet. Then the response value is computed using the MD5 algorithm and matched against the response value sent by the SIP UA. The response value is a hash that depends on the concatenation of all values in the authorization request. If the computed response matches the response sent by the SIP UA, the request is approved (authorized) and normal SIP call flow processing is allowed.

E. User Interface

The bare PC SIP Server has a simple user interface that displays its basic configuration and state information when the interface function sipserverstate() is called. The displayed information includes the number of users added to the username and password database, and the server's configuration mode (proxy, redirector, authentication, stateless, or stateful). The server can also show the username, ip address, and port for each user logged into the system. An administrator can toggle through the list of users, or configure the server so that the display is triggered every time a user is added or removed from the Registered_User_Database by calling sipserverstate() from the Main task.

V. BARE PC SIP UA IMPLEMENTATION

The bare PC SIP user agent (UA) is integrated with the bare PC softphone enabling calls to be set up. Its operational characteristics are similar to those of a SIP UA in a conventional OS-based SIP softphone. However, the UA implementation is different due to the absence of an OS and a built-in protocol stack, and results in a UA with less overhead and better security. The UA can also directly communicate with a peer (without using a SIP server) provided the peer can be contacted via a known (public) destination IP address and port number.

A. UA Operation/User Interface

As in the case of the bare PC SIP server, only two tasks Main and Rcv are needed for the UA, and arriving SIP messages and responses are processed in a single thread of execution as described earlier. When the UA is booted, if an IP address for the UA has not been preconfigured, the UA sends out a request for an IP address and obtains one using DHCP. If this is a private address, the UA is behind a NAT and uses STUN [16] to learn its public IP address and port. In this case, the UA first sends a DNS request and obtains the IP address of a public STUN server. The bare PC STUN implementation is described in more detail below.

After UA completes the initialization process it displays the main login menu, which enables the user to login-in to a particular SIP server or to communicate directly with a peer as noted earlier. In case SIP server login is selected, the UA sends a SIP Register request to the server after performing a DNS resolution if needed. Once the 200 OK messages are received from the SIP server, the UA displays a "main menu" screen as in Fig. 5. The menu has several options, which enables the user to see the IP configuration information from DHCP, and NAT mappings from STUN that show the external IP address and internal/external SIP and RTP ports for the softphone. Such information is useful to troubleshoot connectivity problems. In addition, a separate option shows call status and connectivity information, and indicates whether security is on. A "quick dial" option for selecting specific users is also available.

The software design of the bare PC SIP UA is simple and modular. The essential UA functionality contained in the SIPUA object consists of 3000 lines of C++ code. This object is supplemented by 1) objects for cryptographic and other algorithms needed for key establishment (HMAC, SHA-1, MD5, AES, and Base64); 2) objects implementing the essential elements of the necessary auxiliary protocols (STUN, DHCP, and DNS); and 3) objects needed by the bare PC softphone including the Ethernet, IP, and UDP objects, the RTP, audio, and G.711 objects that handle voice data processing, recording, and playback on the bare PC softphone, and the SRTP [17] object that provides VoIP security.

B. User Agent Client and User Agent Server

The bare UA consists of two independent components: the SIP user agent server (UAS) and SIP user agent client (UAC). The UAS is operationally similar to the bare PC SIP server with respect to its handling of SIP packets. For example, it listens for call requests and its actions are activated by the Rcv task when a packet arrives as discussed earlier for the case of the SIP server. The UAC can be activated by keyboard input. The UA functionality is contained in a SIPUA object that is responsible for processing SIP messages and SDP tags, displaying the SIP UA interface, and interacting with the user. The SIPUA object is integrated in a single AO with several other objects needed to implement the UA.

C. STUN/DHCP/DNS/SRTP

The public IP address and port learned from the public STUN server is used in SIP Invite requests to enable the peer to communicate with the UA behind the NAT. The bare PC SIP UA sends out multiple STUN messages to find the external port for its voice channel over RTP. Since the signaling channel is proxied through the SIP server, STUN

 ISBN: 978-1-61208-107-6

is not needed to discover the external SIP signaling port. After the bare PC client is booted, STUN messages for the media channel are sent every 30 seconds until the SIP UA establishes the call. The Invite message contains the last known media channel external port number. Since the NAT binding may change, the UA sends voice packets to the destination host using a sequence of consecutive ports. The UA stops sending on the other ports once voice packets are received on a particular port.

Since there is no OS and no built-in protocol stack on the bare PC softphone, the bare PC SIP UA also needs to send DHCP messages to automatically obtain an IP address and other essential configuration information at start-up. The DHCP messages follow the typical DHCP call flow (Discover, Offer, Request, and Ack). The softphone can also send DNS requests to resolve the domain name of the SIP or STUN server. As noted earlier, the implementation of the DHCP and DNS protocols have only the minimal features needed by the bare PC SIP softphone.

The bare PC SIP UA is also integrated with SRTP. The implementation and performance of SRTP on a bare PC softphone are presented in [18]. SRTP allows the UA to communicate securely with conventional SIP UAs that are SRTP capable. The bare PC softphone AO includes implementations of SHA-1, MD5, HMAC, and AES in counter mode, which are used by SRTP. The bare PC SRTP implementation also supports addition of a recommended authentication tag to the end of the RTP packet. The UA currently implements the SDP Offer/Answer model via SDES for key exchange. This method is used by several conventional SRTP clients. The keys used to generate the session keys are Base64 encoded by the bare PC softphone SRTP implementation prior to transmission. Since this approach for transmitting keys is not secure, TLS is used by some conventional softphones for SIP signaling.

VI. TESTING

Operational tests of the bare PC SIP server and SIP softphone implementations with and without authentication and SRTP security were conducted using Dell GX-260 desktops with Intel Pentium 4 2.4 GHz processors, 1.0 GB RAM, and a 3COM Ethernet 10/100 PCI network card. The test network consists of a dedicated LAN within the Towson University network, and an external network connected through an ISP as shown in Fig. 6. The bare PC SIP server and user agents were first tested within the dedicated LAN. Testing was performed to verify 1) correct operation between the bare PC SIP server and bare PC SIP softphones; 2) interoperability of bare PC SIP softphones with the OpenSer v3.0.0 server [19]; 3) interoperability of the bare PC SIP server with Snom360-5.3 softphones [20]; and 4) interoperability of bare PC SIP softphones with the Snom softphones.

Similar tests were conducted over the Internet by establishing calls between a softphone on the external network and another on the dedicated LAN when the SIP servers are connected to the LAN. These tests also served to verify that the UA and the lean DHCP, STUN, and DNS implementations on the bare PC SIP softphone work correctly when it is connected to the Internet. In particular, the bare PC STUN implementation was found to be adequate for connecting between clients behind NATs on the dedicated test LAN and on an ISP network.

VII. CONCLUSION

We described the design, implementation, and operations of a bare PC SIP server and SIP UA, which provide essential SIP functionality with less overhead and better security at lower cost due to the absence of an OS. The underlying bare PC system enables the SIP server and UA to benefit from simple tasking, lean protocols, and efficient data handling. The tests conducted show that the bare PC SIP server can interoperate with both bare PC and OS-based UAs, and also that the bare PC SIP UA can interoperate with both an OS-based UA and an OS-based SIP server.

REFERENCES

[1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, “SIP: Session Initiation Protocol,” RFC 3261, 2002.

[2] L. He, R. Karne, and A. Wijesinha, “The Design and Performance of a Bare PC Web Server”, International Journal of Computers and Their Applications, vol. 15, pp. 100 - 112, June 2008.

[3] G. Ford, R. Karne, A. Wijesinha, and P. Appiah-Kubi, “The Peformance of a Bare email server”, 21st International Symposium on Computer Architecture and High Performance Computing, SBAC-PAD 2009, October 28-31, Sao Paulo, Brazil, pp.143-150, 2009.

[4] L. Chen, and C. Li, “Design and Implementation of the Network Server Based on SIP Communication Protocol,” World Academy of Science, Engineering and Technology 31, pp. 138-141, 2007.

[5] S. Zeadally and F. Siddiqui, “Design and Implementation of a SIP-based VoIP Architecture,” AINA 2004.

[6] A. Singh, A. Acharya, P. Mahadeva, and Z-Y, Shae, “SPLAT: a unified SIP services platform for VoIP applications,” International Journal of Communication Systems, Volume 19, Issue 4, pp. 425-444, 2006.

[7] P. Zave, E. Cheung, G. W. Bond, and T. M. Smith, “Abstractions for Programming SIP Back-to-Back User Agents,” IPTComm’09, 2009.

[8] S Siddique, RK Ege, SM Sadjadi, “X-Communicator: Implementing an advanced adaptive SIP-based User Agent for Multimedia Communication,” SoutheastCon, pp. 271-276, 2005.

[9] K. J. Kim, Y, Jang, J. W. Chung, and J. H. Seo, “Design and implementation of SIP UA for a manufacturing network,” International Journal of Advanced Manufacturing Techniques, Volume 28, Number 7-8, pp. 822-826, 2006.

[10] K. Singh and H. Schulzrinne, “Peer-to-Peer Internet Telephony using SIP,” International Workshop on Network and Operating System Support for Digital Audio and Video, pp. 63-68, 2005.

[11] A. Alexander, A. L. Wijesinha, and R. Karne, “A Study of Bare PC SIP Server Performance,” 5th International Conference on Systems and Network Communications, ICSNC 2010, In Press.

[12] G. H. Khaksari, A. L. Wijesinha, R. K. Karne, L. He, and S. Girumala, “A Peer-to-Peer bare PC VoIP Application,” IEEE Consumer and Communications and Networking Conference (CCNC 2007), pp. 803-807, 2007.

[13] G. H. Khaksari, A. L. Wijesinha, R. Karne, Q. Yao, and K. Parikh, “A VoIP Softphone on a Bare PC”, Embedded Systems and Applications Conference (ESA), 2007.

[14] R. K. Karne, K. V. Jaganathan, T. Ahmed, and N. Rosa, “DOSC: Dispersed Operating System Computing,” 20th Annual ACM Conference on Object Oriented Programming, Systems, Languages, and Applications (OOPSLA ’05), Onward Track, pp. 55-61, 2005.

[15] R. K. Karne, K. V. Jaganathan, and T. Ahmed, "How to Run C++ Applications on a Bare PC?" 6th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 2005), pp. 50-55, 2005.

[16] J. Rosenberg, R. Mahy, P. Matthews, and D. Wing, "Session Traversal Utilities for NAT (STUN)," RFC 5389, 2008.

[17] M. Baugher, D. McGrew, M. Naslund, E. Carrara and K.Norrman, "The Secure Real-time Transport Protocol (SRTP)," RFC 3711, 2004.

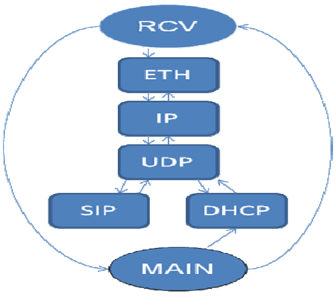


Figure 1. SIP Server Protocol/Task Relationships.

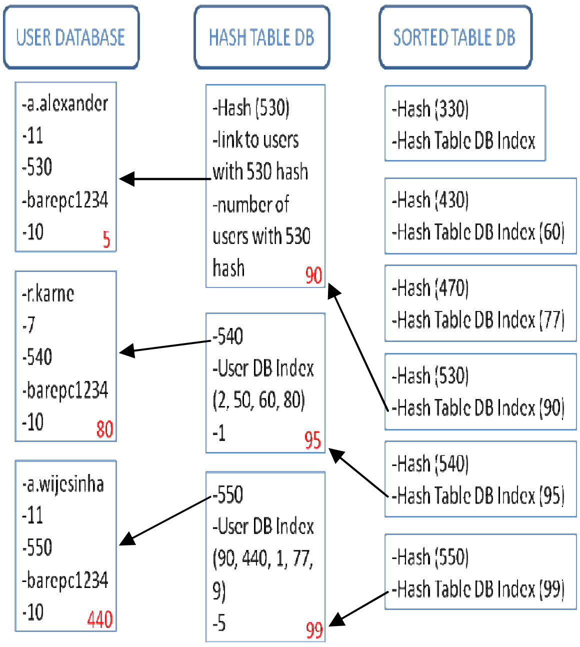


Figure 2. Database and Hash Table Relationships.

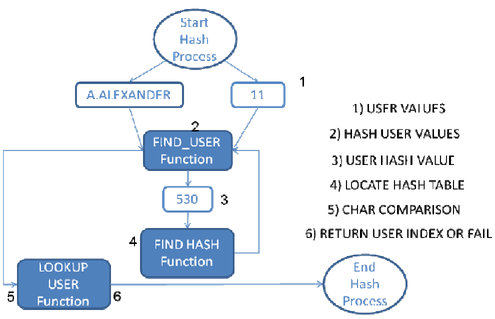


Figure 3. User Lookup Process.

[18] A. Alexander, A. L. Wijesinha, and R. Karne, "An Evaluation of Secure Real-Time Protocol (SRTP) Performance for VoIP," 3rd International Conference on Network and System Security (NSS), pp. 95-101, 2009.

[19] Kamailio (OpenSER) SIP server, http://sourceforge.net/projects/openser

[20] Snom VoIP phones, http://www.snom.com/download/snom360-5.3.exe


```
INVITE sip:67890111@barepc.towson.edu:5060 SIP/2.0
Via:SIP/2.0/UDP192.168.1.56:5060;brach=0320
From:<sip:0123456@ barepc.towson.edu>;tag=0
To: <sip: 67890111@ barepc.towson.edu>
Max-Forwards: 70
Call-ID: 0010-0003-DA76506F-0@AAE2A42DF82D1D0AA
 CSeq: 297386 INVITE
Contact: <sip:123456@192.168.1.56:5060>
Content-Type: application/sdp
Proxy-Authorization:Digest
username="8000",realm="BAREPC",nonce="3bd76584",
uri="sip:123456@192.168.2.81",response="6e91de67ad976997
ff"
User-Agent: BarePC SIP UA v1.0
Content-Length: 276
v=0
 o=Vega400 4 1 IN IP4 192.168.1.56
 s=Bare PC Sip Call
 t=0 0
 m=audio 10006 RTP/AVP 4 18 8 0 96
 c=IN IP4 192.168.1.56
 a=rtpmap:8 PCMA/8000
 a=rtpmap:0 PCMU/8000
 a=rtpmap:96 telephone-event/8000
 a=fmtp:96 0-15,16
 a=sendrecv
```

Figure 4. SIP Invite with Authentication.

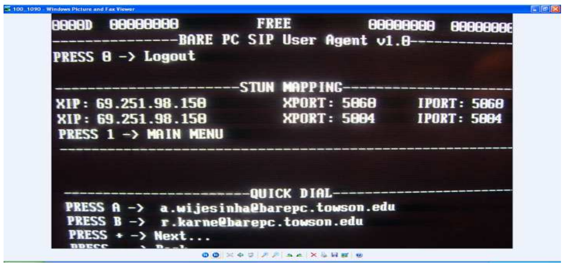


Figure 5. UA Main Menu Screen.

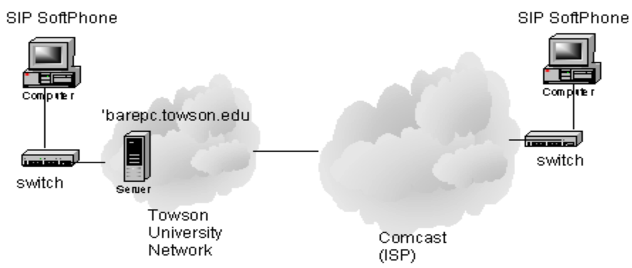


Figure 6. Test Network.

A Dendritic Cell Inspired Security System in Wireless Sensor Networks

Jingjun Zhao
Department of Computer Science
North Dakota State University
Fargo, ND, USA
jingjun.zhao@ndsu.edu

Kendall E. Nygard
Department of Computer Science
North Dakota State University
Fargo, ND, USA
Kendall.Nygard@ndsu.edu

***Abstract* – We describe a scalable distributed methodology for increasing the rate of real packets received by the base station (BS) in a wireless sensor network (WSN) and to limit the inimical impacts of intruders in the network. The proposed security mechanism adopts a dual protection scheme to ensure that the BS obtains maximal real packets. First, we utilize a Dynamic Dendritic Cell Algorithm (DDCA) that effectively detects harmful intruders in a WSN and dynamically adjusts the monitoring period in response to the situation of the network. A sensor node running this algorithm can identify fake packets generated by the intruders based on pre-defined rules. Second, we apply a Markov Chain Monte Carlo (MCMC) method called the Metropolis-Hastings (MH) algorithm to infer the location of intruders in a wireless sensor network using partial information obtained from a subset of the sensor nodes. In turn these inferred locations are used by a fuzzy logic algorithm that we apply to assess the effect of the intruders on a monitored point. Based on the assessment, the BS sends commands that adjust the monitoring period that a sensor node uses to identify an intruder. The mechanism increases the flexibility and accuracy of the DC-inspired algorithm in accurately identifying harmful intruders, especially for harmful mobile intruders. The method can be applied to different sizes of WSNs and to both dynamic and static WSNs. We simulated the proposed algorithms using JADE (Java Agent Development Framework), and the results demonstrate good performance.**

Keywords-Wireless Sensor Networks; Dendritic Cells; Metropolis-Hastings algorithm; Fuzzy Logic; JADE

I. INTRODUCTION

A wireless sensor network (WSN) is composed of large numbers of small devices called sensor nodes that are typically deployed in an open and unprotected environment. They collect information and transmit data packets to a Base Station (BS). WSNs have been widely used in military and civil applications such as battlefield monitoring, environment and habit monitoring, and factory automation management. However, a WSN is susceptible to attacks and sensor failures such as packet dropping, packet change, energy-exhaustion, etc. Intrusion detection is an important research topic in a WSN, to help decrease power loss and to increase malicious event detection. Intrusion Detection methods utilized in wired networks are difficult to apply directly to WSNs because the sensor nodes are limited in battery power, storage, and computational ability. An Artificial Immune System (AIS) is a problem-solving methodology inspired by how biological immune systems in mammals are able to detect pathogens and destroy them before they cause harm to the body.

More specifically, Negative Selection Algorithm (NSA) has been used for solving different anomaly detection problems [14]. A NSA uses a learning phase to construct detectors which can identify and dispatch invaders, but are not harmful the organism itself. A fundamental issue in a NSA is that very difficult to maintain complete "non-self" detection in many real applications [13].

Following another type of AIS, the work in [4] advanced the Danger Theory (DT) approach to intrusion detection. Subsequent work by Nauman and Muddassar [6] established a security system based on the Dendritic Cells behavior. Work reported in [9] included detailed rules for the Dendritic Cell Algorithm (DCA) for analyzing abnormal signals. These DCAs are more flexible at detecting misbehaviors than NSAs, but do require monitoring period to identify an intruder. The period is fixed at the initial stage of a WSN and there is a tradeoff in deciding an appropriate monitoring period. A large period may lead to a low detection rate, and a little period may lead to a high error rate.

In our work, we employ a dual protection model that dynamically–adjusts to detect both static and mobile intruders while maintaining low energy consumption. Our DCA is primarily used to detect attacks: packet change, fake packet and energy-exhaustion. The ability to defend against these basic but widely existing types of attacks in a WNS makes our DCA a good fit in practice. To maximize the detection ability of our DCA, we also utilize the Metropolis-Hastings Algorithm (MHA) and fuzzy logic to dynamically adjust the monitoring period used in the DCA. The BS collects partial information about an intruders' location by requesting from a subset of the sensor nodes, then applies the MHA to estimate the location of the intruder. Because only a subset of the sensor nodes transmits location information, network longevity is enhanced. Upon acquiring the inferred information, the BS implements a fuzzy logic assessment algorithm that assesses the effect of these intruders on a monitored point. The assessment is sent back to the sensor nodes in this area, and these sensor nodes use it to adjust their monitoring period accordingly. This dynamic-adjusting mechanism ensures that the proposed DCA can accurately identify intruders and thus increases event detection reliability.

The rest of the paper is organized as follows: Section II discusses the related work; Section III describes the proposed DC-inspired algorithm, the usage of the Metropolis-Hastings algorithm, and the Fuzzy Logic algorithm; Section IV describes the designed multi-agent architecture for wireless sensor networks; Section V details our implementation and simulation results; finally, the conclusion and future work are given in Section VI.

 ISBN: 978-1-61208-107-6

II. RELATED WORK

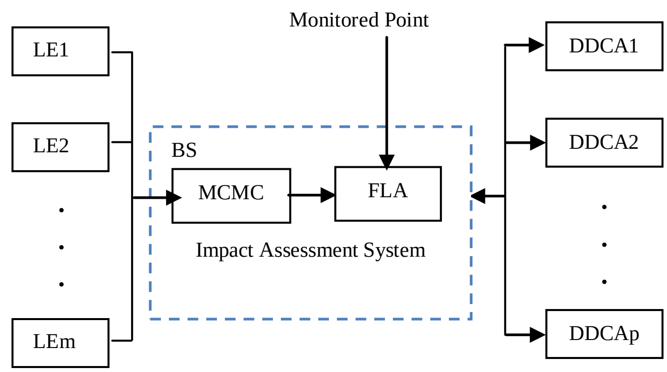
Greensmith, Aickellin and Cayzer [5] proposed a DC-based algorithm for the detection of anomalies. The authors categorize signals as Pathogen Associated Molecular Patterns (PAMPs), Safe Signals (S), Danger Signals (D), or Inflammatory Cytokines (IC). In [1], a description of the similarity between WSNs and AIS is provided, showing that a Dendritic Cell algorithm (DCA) can detect Cache Poisoning attack in a WSN. In this work, each node has an interest cache with fixed size that is used to record received history interest packets. Newly received packets replace older ones in the interest cache when the cache is full. These history packets are used to help a node decide whether or not to drop a received packet. The Ubiquitous Dendritic Cell algorithm (UDCA) combines the interest cache and the data cache in a node to analyze and identify danger signals. This algorithm can potentially detect an interest cache poisoning attack at an early stage. In [4], the authors employ Danger Theory (DT) in an intrusion detection system. Nauman and Muddassar [6] built upon an existing AIS-based security system described in [7] by simulating the behavior of a DC. In this approach, danger signals lead to the maturing of the DCs. The matured DCs activate detectors, which are then used to detect "non-self" network nodes. The work in [8] introduced a Dendritic Cell Algorithm. In [9], rules are formulated for identifying intrusion in a WSN.

The fixed cache size and monitoring period assumed in these algorithms will restrict their effectiveness in some applications. For example, these algorithms will have less efficiency on detecting mobile intruders than detecting static intruders in a WSN because mobile intruders can move out from the surveillance range of a sensor node before being detected. In this paper, we propose a Dendritic Cell algorithm that dynamically adjusts the monitoring period in a WNS. Our algorithm has great flexibility when compared with these existed algorithms.

III. THE DENDRITIC CELL APPROACH

A. Dendritic Cells

In a biological immune system, Dendritic Cells (DCs) are considered to be the most important Antigen Presenting Cells (APCs). Their basic role is to mark the surface of harmful antigens so that they can be recognized and destroyed by other cells in the immune system. The DCs are derived from bone marrow cells and begin their existence in an immature state. As an immature DC collects surrounding antigens, it is transformed to a semi-mature state. Eventually a semi-mature DC can receive sufficient danger signals from antigens to transform to a fully mature state at which time it is capable placing the danger marks on the surface of the harmful antigens. A Dendritic Cell Algorithm (DCA) is a problem-solving method that is based on the behavior of dendritic cells. We have designed a new DCA used to detect harmful intruders in a WSN. An innovation in our method is that the, monitoring period is dynamically adjusted by the base station in accordance with the current network status. Thus, we refer to the method as a Dynamic Dendritic Cell Algorithm (DDCA). To achieve the dynamic behavior, we employ a Markov Chain Monte Carlo (MCMC) technique to infer the distribution of the intruders in the network, then apply a Fuzzy Logic Algorithm (FLA) to assess the impact of these intrudes on a monitored point. Figure 1 shows the integration of the DC-inspired detection system and impact assessment system. We describe these three algorithms in more detail in the following sections.



m: the number of sensor nodes that are queried by the BS
p: the number of sensor nodes around the monitored point
LEi: the location of intruder i
DDCAi: the sensor node i running DDCA around the monitored point

Figure 1. The block diagram showing the integration of Dynamic Dendritic Cell Algorithm (DDCA), MCMC and Fuzzy Logic based Algorithm (FLA)

B. The Dynamic Dendritic Cell Algorithm (DDCA)

In this paper, we primarily recognize the following types of attacks in a WSN:

- the transmitting of changed messages;
- the reporting of messages at a frequency different than normal or expected ; and
- the reporting of fake messages.

When a sensor node sends out a message, we assume that all the neighbor nodes of the sender can receive and interpret the message. Each sensor node has the ability to discern if a neighbor node is transmitting a suspicious message (such as a changed or false message). A sensor node flags a harmful intruder if the number of suspicious message exceeds a pre-specified threshold value.

Sensor nodes are typically battery powered and have limited energy that must be used efficiently. The energy consumption rate for communication greatly exceeds that for sensing. Thereby, the useful lifetime of a WSN is largely governed by the management of the transmitting and receiving of messages. Ignoring messages rather than communicating with harmful intruders is energy conserving.

In the first phase of identifying a harmful intruder, suspicious messages are flagged by placing the senders onto a semi-harmful intruder (SHI) list. Additional detailed monitoring of subsequent message traffic will trigger the placing of the node onto a harmful intruder (HI) list. If an innocent period has elapsed, a node is deleted from the SHI list. Figure 2 illustrates the structure of this algorithm.

 ISBN: 978-1-61208-107-6

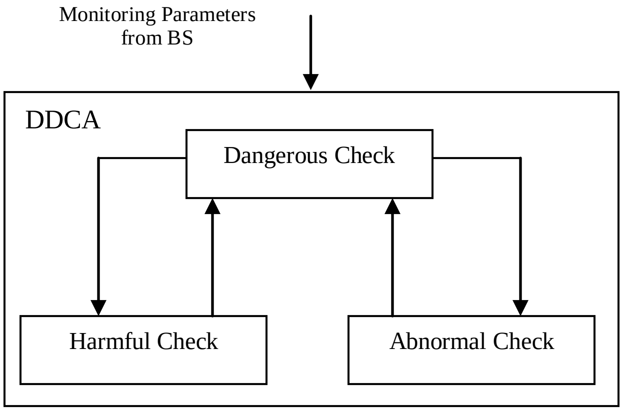


Figure 2. The Structure of Dynamic Dendritic Cell Algorithm (DDCA)

The following code in Figure 3 describes the DDCA in detail.

---

m_n: new message
HI: Harmful Intruder
SHI: Semi-Harmful Intruder
T_1: the maximum time slice calculating abnormal frequency of reporting message
T_2: the maximum time slice supervising a suspicious intruder
t_0: threshold for reporting message frequency during T_1
t_1: threshold for reporting message frequency during T_2
t_2: threshold for reporting false message rate during T_2
t_3: threshold for reporting changed message rate during T_2
(Note: to tolerate casual high frequency of reporting message, let $T_2 > T_1$)

//To check if the received message is dangerous
BOOL **CheckDangerMessage** (m_n)
  IF (m_n came from HI)
    return TRUE
  ELSE IF (m_n came from SHI)
    IF (CheckHarmfulIntruder(m_n) == TRUE)
      return TRUE
  ELSE IF (CheckAbnormalMessage(m_n) == TRUE)
    update SHI list
  ELSE
    return FALSE

//To check if the received message is abnormal
BOOL **CheckAbnormalMessage** (m_n)
  IF (m_n is a changed original message)
    return TRUE
  ELSE IF (m_n is a fake message)
    return TRUE
  ELSE IF (the frequency of the sender reporting message $> t_0$)
    return TRUE
  ELSE
    return FALSE

//to decide if the message sender is an harmful intruder by checking the history records
BOOL **CheckHarmfulIntruder** (m_n)
  update the history message record of the sender
  estimate the frequency and percentage of the sender
  IF ((the new frequency $> t_1$) **or** (the new percentage of false message $> t_2$) **or** (the new percentage of changed message $> t_3$))
    clear the history record of the sender
    move the sender from SHI list to HT list
    return TRUE
  ELSE IF (the supervised time of the sender $> T_2$)
    remove the sender from SHI list
  ELSE
    return FALSE

Figure 3. The Dynamic Dendritic Cell Algorithm (DDCA)

This algorithm consists of three sub-functions: checking for dangerous messages, abnormal messages and harmful intruders. Figure 2 presents the relationships among these sub-functions. When a sensor node receives a new message, the algorithm first checks the HI list. If the sender of this message is an identified harmful intruder, the message is considered to be a dangerous message. If the sender is an identified suspicious intruder that exists in the SHI list, the algorithm runs the function for identifying harmful intruder to decide if this sender is dangerous enough to be considered harmful. Finally, the algorithm runs the function of checking for abnormal message and possibly puts the sender on the SHI list. This algorithm will only consume limited energy of a sensor node because of the simple calculation in each sub-function. The dynamic adjustment of parameters increases the flexibility and accuracy of the DC-inspired algorithm in detecting dangerous messages.

C. *The MCMC Method*

We employ a Markov-chain Monte-Carlo (MCMC) method to infer the location of an intruder. The MCMC approach is based upon the Metropolis-Hastings algorithm. Figure 4 describes the general process of this algorithm. By recognizing the location of intruders, the BS can do a more effective assessment of received packets. If the BS were to send query messages to all sensor nodes in the WSN to collect location information, excessive energy would be required. The MCMC technique infers the intruders' location from limited local information. We assume that the fixed intruders are uniformly distributed over the network in the beginning. To the proposal distribution, we let the random location of an intruder moving to is chosen uniformly. That means the movement of an intruder is conditionally independent, which is a necessary condition for running the Metropolis-Hastings algorithm. In the training phase, a sensor node reports a message to the BS when an intruder is identified. The BS obtains the intruders' initial location from these messages.

Initial x_0

For i = 0 to n-1

Sample $u \sim u[0,1]$

Sample $y \sim q(y|x_i)$

$\alpha\,(x_i, y) = \min\{1, \frac{\pi(y)q(x_i|y)}{\pi(x_i)q(y|x_i)}\}$

If ($u <$ $\alpha\,(x_i, y)$)

$x_{i+1} = y$

Else

$x_{i+1} = x_i$

Figure 4. The Metropolis-Hastings algorithm

Figure 4 shows that a proposal movement is accepted if the calculated probability α bigger than a random number μ uniformly drawn from between 0 and 1. To minimize the complexity of the calculations, we use the method of [12] shown below for the acceptance criterion α.

$$\alpha\,(LE^t, LE^{t+1}) = \min\,(1, \frac{P\,(LE^{t+1}, R_1 \ldots R_P\,)}{P\,(LE^t, R_1 \ldots R_P\,)} \qquad (1)$$

LE^t: the vector of the intruders' location at time t
R_i: the ith sensor measurement
p: the number of nodes that the BS requests

Simplifying the following expression

$$\frac{P\,(LE^{t+1}, R_1 \ldots R_P\,)}{P\,(LE^t, R_1 \ldots R_P\,)} \qquad (2)$$

to

$$\prod_{k=1}^{K} \frac{P(R_{K_k}\,|LE^{t+1})}{P(R_{K_k}\,|LE^t)} \qquad (3)$$

K: the number of affected sensor nodes
K_k: the index of the kth affected sensor

In equation (3), there is only one variable in vector LE that has changed value from step t to step t+1. This change only affects those sensor nodes that can sense the change. This equation makes the calculation of the acceptance criterion easy and fast. The detail process of simplification can be found in [12].

D. The Fuzzy Logic Algorithm

Once the location of the intruder is known, a fuzzy logic algorithm is used to calculate the impact of these intruders on the accuracy of the packets received by the BS from designated areas in the network.

1) Inputs and outputs

a) The fuzzy algorithm has three inputs

Each input is classified into three categories: small, middle and large. The threshold values can be adjusted according to the requirements of different applications.

- Distance – the distance between an intruder and the monitored point
  Small: distance ≤ 5m
  Middle: 5m< distance < 15m
  Large: distance ≥ 15m
- Danger degree – classifying intruders based on their danger attacking type, i.e.,
  Small: reporting false message;
  Middle: transmitting changed original message;
  Large: reporting message in a frequency higher than expected one;
- Relative Position – using the angle of an intruder, the monitored point and the BS to represent the relative position of an intruder and the BS
  Small: $\theta \leq 20^0$
  Middle: $20^0 < \theta < 45^0$
  Large: $\theta \geq 45^0$

b) The fuzzy algorithm has five outputs:

(0, 0.25, 0.5, 0.75, 1)

These outputs represent different impacts of the intruders on the monitored point.

2) The membership functions

Figures 5 and 6 present the member functions of the fuzzy logic algorithm. The outputs of membership functions are defined as: small, middle and large, depending on the distance and the relative position individually.

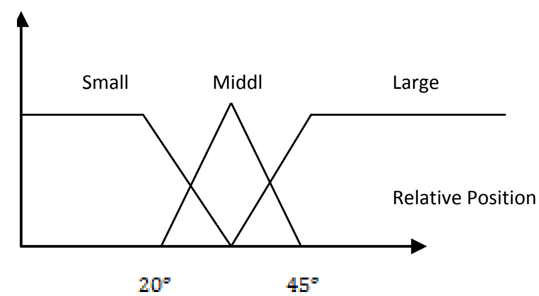


Figure 5. The input is the relative position.

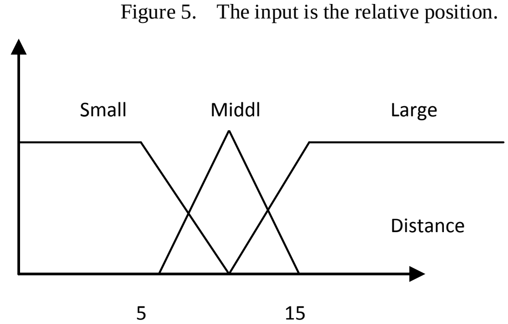


Figure 6. The input is the distance.

3) Fuzzy Rules

The fuzzy rules are listed in Table I. Shorter distance, smaller relative position, and consuming more energy will result in a classification of danger.

TABLE I. FUZZY RULES

Inputs			Output
Distance	*Danger Degree*	*Relative Position*	
S	M/S	M/S	0.25
S	M/L	S	0.5
S	L	L	0.75
S	L	M/S	1
M	M/S	M/L	0.25
M	L	M	0.5
M	L	S	0.75
L	-	M/L	0
L	-	S	0.25

4) Usage of the Fuzzy Logic Algorithm

The fuzzy logic algorithm is used to assess the impact of the intruders on a monitored point in a sensor network. To calculate the impact, we define the following equation:

$$\overline{IA} = \frac{\sum_{i=1}^{n} IA_i}{N}$$

$\overline{IA}$: the assessed impact of intruders

IA_i: the impact of the ith intruder (i.e., the output of the fuzzy logic algorithm)

n: the number of intruders

N: the number of sensor nodes around the monitored point

The BS in a WSN will adjust the monitoring parameters T_1 and T_2 used in the DDCA algorithm, according to the new assessment. Because the value N is fixed, more intruders will increase the assessed impact. Our rule is that the bigger the assessment is, the smaller the monitoring parameters become.

IV. MULTI-AGENT ARCHITECTURE

Using Multi-agent Systems (MASs) to model WSNs is considered to be a powerful and flexible approach [11]. The work reported in [10] concerns implementing a MAS using JADE (Java Agent DEvelopment Framework), a software framework fully implemented in JAVA. Similarly, we use JADE for our work. We have identified a set of common functions in WSNs and map each function with an agent, as shown in Table II. The architecture is composed of five types of agents: the Central Control Agent (CCA), the Sensing and Transmitting Message Agent (STMA), the Environment Agent (EA), the Message Analysis Agent (MAA), and the Immune System Agent (ISA). Figure 7 illustrates the designed Multi-Agent Architecture.

TABLE II. MAPPING BETWEEN AGENTS AND FUNCTIONS IN A WSN

Agents	Functions in a Wireless Sensor Networks (WSN)
The Central Control Agent (CCA)	Control center in a WSN
The Sensing and Transmitting Message Agent (STMA)	A sensor node
The Environment Agent (EA)	Environment of sensor nodes deployed
The Message Analysis Agent (MAA)	Sense of abnormal messages
The Immune System Agent (ISA)	Intruder detection

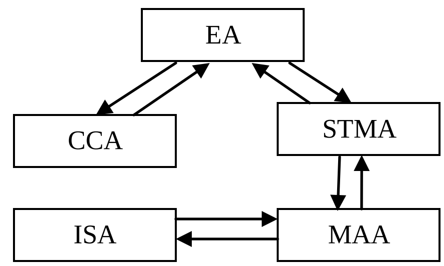


Figure 7. The designed Multi-Agent Architecture

The CCA sends control commands to the STMA. The STMA collects information and sends it to the CCA. The EA provides an operating environment for the other agents. The MAA is used to judge if a node is a potentially harmful intruder. The ISA is responsible for recognizing harmful intruders. The ISA is activated only when a message is recognized as a suspicious message by the MAA. The detailed functions of the agents are listed in Table III We run the proposed algorithm in the ISA to discriminate between normal messages and abnormal messages.

TABLE III. AGENTS' RESPONSIBILITIES

Agent type	Responsibilities
The Central Control Agent (CCA)	Sends commands to the EA Receives messages from the EA
The Sensing and Transmitting Message Agent (STMA)	Sends messages to the EA. Presents incoming messages from the EA Allows a sensor to accept incoming messages Allows a sensor to reject incoming messages
The Environment Agent (EA)	Presents incoming messages from the CCA and the STMA Sends the received messages to the CCA and the STMA
The Message Analysis Agent (MAA)	Sends messages to the STMA Sends messages to the ISA Recognizes abnormal message
The Immune System Agent (ISA)	Sends messages to the MAA Recognizes harmful intruders

V. SIMULATION RESULTS

We evaluate the performance of the security scheme using simulation. We implement the algorithms in JADE, which is an agent platform that is compliant with the Foundation for Intelligence Physical Agents (FIPA) standards for multi-agents. Our multi-agent architecture is similar to that reported in [10], which supports differing types of agents and inter-agent communication. Each sensor node agent hosts the DC-inspired algorithm, and the BS agent hosts the MH algorithm and the fuzzy logic algorithm.

The test area is a 100 by 100 square region, and 100 sensor nodes are randomly deployed in this area. Our experiments use a fixed number of intruders randomly deployed in the test area.

We conducted three groups of experiments to evaluate the efficiency of the dual protection scheme in a sensor network. The first group of experiments is executed to estimate the efficiency of the proposed DC-inspired algorithm on maintaining the network lifetime. The second group of experiments is performed to measure the effect of cache size on the intruder detection rate. The last group of experiments is conducted to assess the performance of the dual protection scheme on detecting mobile intruders in a WSN. The intruders in the sensor network are static in the first two groups of experiments and mobile in the last one. The total number of sensor nodes is constant in all experiments.

A. *Maintaining the network lifetime*

In this group of experiments, there are 100 sensor nodes in the test area. When a sensor node runs out of battery life, normally or abnormally, it becomes a failed sensor node. We assume that a prescribed number of failed nodes will cause network failure and treat this as the only factor affecting the network lifetime. We set the number of failed nodes that cause the network failure to 70. To estimate the efficiency of the DC-inspired algorithm, we sample the values of failed sensor nodes at different time slots, and compare the results with an experiment that simulates our DC-inspired algorithm and a parallel experiment in which no security mechanism is employed. We randomly deploy 10, 20 and 30 intruders in the test area and perform experiments on each case. The results show that the DC-inspired algorithm greatly increase the network lifetime when compared with no security mechanism employed. Figure 8 shows that the network lifetime is doubled. Figures 9 and 10 show a longer network lifetime despite more intruders added. It is related to the final distribution of intruders.

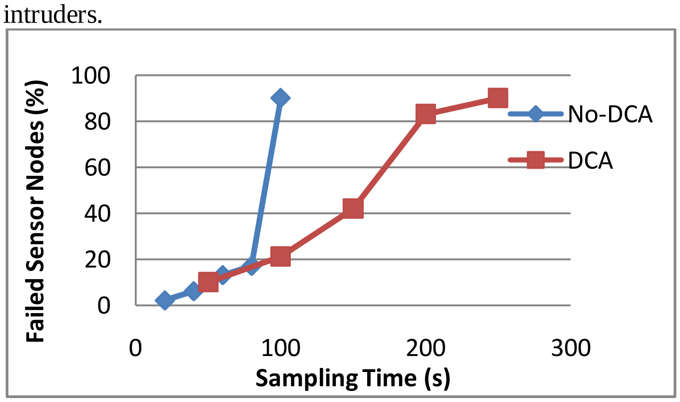


Figure 8. Sampling Time vs Failed rate of 10 intruders

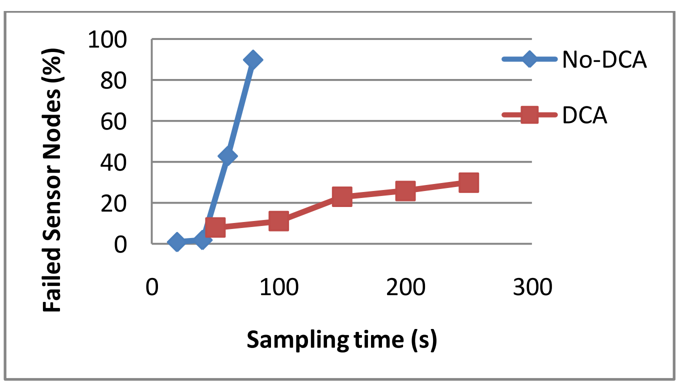


Figure 9. Sampling Time vs Failed rate of 20 intruders

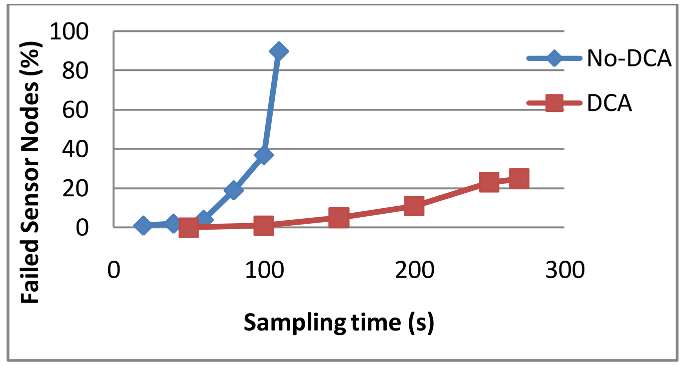


Figure 10. Sampling Time vs Failed rate of 30 intruders

B. *Cache Size Effect on the Detection Rate*

In the second group of experiments, we keep the same size of the test area and the network size as in the first group of experiments. Our aim is to measure the effect of the cache size on the detection rate of the DC-inspired algorithm. To detect a harmful intruder, a sensor node uses a cache to store the abnormal packets received from an intruder, and to monitor the intruder for a period of time to assess whether or not this is a harmful intruder. We set the cache size to 50, 100, 200, 500 and 1000 units and perform experiments on each cache size.

Figure 11 shows the experimental results. As expected, a larger size cache uniformly has a higher detection rate than a small cache. There is a flattening of the curves after a cache size of 100, indicating only marginal benefits of caches of size 200 or larger. When there are 10 intruders, the algorithm detect almost all of the harmful intruders with a cache size of 1000. The results also show that the procedure achieves a uniformly better detection rate for smaller numbers of intruders than for larger numbers, for any of the cache sizes. This is because the packets that a sensor node receives from harmful intruders are saved in the shared cache, and, when the cache is full, old packets will be removed from the cache, even though these packets may still be useful for detecting an intruder. Hence, the algorithm has a lower detection rate when more intruders exist in the network. These experiment results indicate that cache size is an important factor for detecting harmful intruders. However, large cache sizes may be unrealistic in resource-limited low-end sensors.

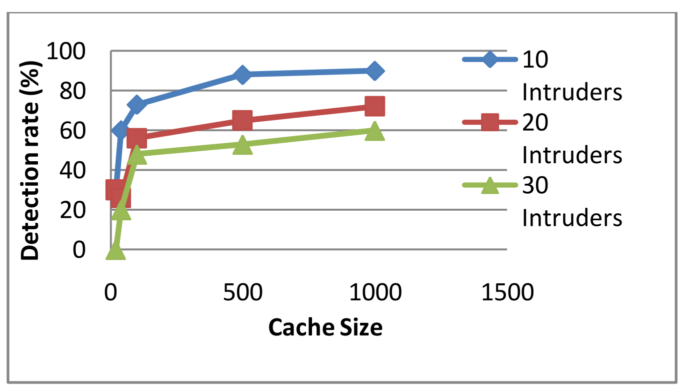


Figure 11. Cache Size vs Detection Rate

 ISBN: 978-1-61208-107-6

C. Assessment of the Dual Protection Scheme

The third group of experiments is designed to assess the performance of the proposed dual protection scheme. In these experiments, a fixed number of mobile intruders are randomly deployed in the sensor network. We measure the accuracy of the received packets by the BS to evaluate the efficiency of the scheme. At the beginning of the experiments, a monitored point is randomly chosen in the test area. In a real application, the BS could specify the monitored point according to the actual network topology. In our experiments the BS knows the number of intruders and their initial location, which is necessary condition in the MH algorithm. In a real application, the BS can obtain this information from the received packets in the startup phase of the sensor network.

The number of mobile intruders we choose in the experiments varies from 5 to 30 in step sizes of 5. These mobile intruders are randomly deployed in the initialization phase of the sensor network. Each mobile intruder picks a random direction to move at the speed of 5 units per second. Figure 12 shows the experiment results for detecting both static and mobile intruders. The results show that mobile intruders reduce the accuracy of the detection.

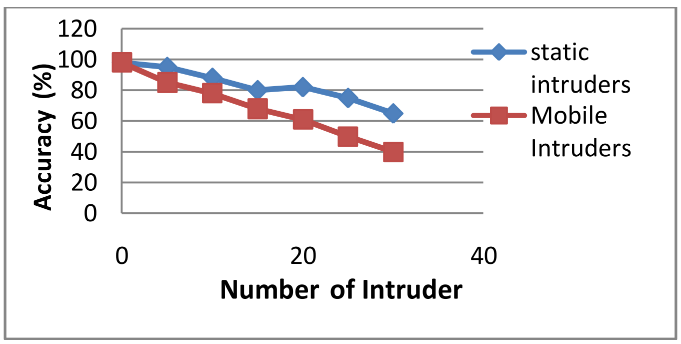


Figure 12. DC-inspired algorithm detecting static and mobile intruders

Figure 13 shows the simulation results of our dual protection scheme for a WSN. It is clear that the BS can obtain more accurate packets by using the dual protection scheme than by only performing the standard DC-inspired algorithm. The dual scheme effectively reduces the impact of the mobile intruders on the DC-inspired algorithm, and this advantage is more obvious when there are more mobile intruders in the network. When there are more mobile intruders in the sensor network, the chosen monitored point is more likely to be surrounded by the intruders. The BS estimates the impact of these intruders on the monitored point and then sends out the assessment to the sensor nodes around the monitored point. These sensors nodes effectively adjust their monitoring period according to the received assessment. This mechanism improves the detection rate to some extent.

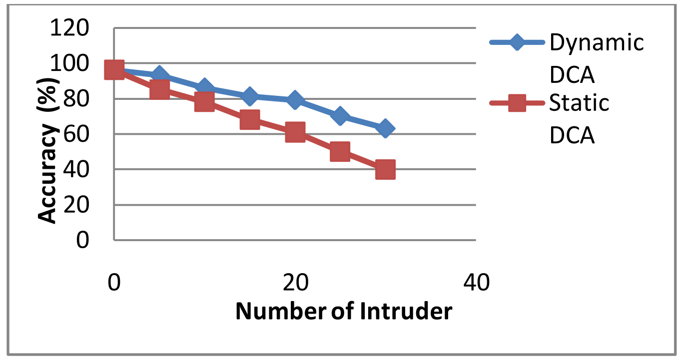


Figure 13. Dynamic DCA vs static DCA on detecting mobile intruders

In summary, our DC-inspired algorithm effectively detects static intruders in a WSN, but mobile intruders can elude detection. The MCMC technique and the fuzzy logic algorithm lower the effect of the mobile intruders and render the algorithm suitable for WSNs with either static or mobile intruders.

VI. CONCLUSION AND FUTURE WORK

We have presented a dual security mechanism for WSNs. A sensor node performing the dendritic cell-inspired algorithm effectively detects harmful intruders that report fake messages, transmit changed packets or report messages with unexpected frequencies. To enhance the ability of the dendritic cell algorithm to detect mobile harmful intruders, we developed an impact assessment system. The enhanced system uses the Metropolis-Hastings algorithm to infer the location of intruders based on partial information, followed by a fuzzy logic algorithm for assessing the impact of intruders on a monitored point. The assessment directs a sensor node to dynamically adjust the monitoring period according to the current network situation. Our simulations demonstrate that the dual protection mechanism promptly and effectively detects static or mobile intruders, extends sensor network lifetime, and improves the accuracy of the packets received by the BS.

In many real applications, such as battlefield monitoring, the number of intruders in a monitoring area will dynamically change. To apply to this kind of application, our security system would need an enhanced MCMC algorithm for tracking multiple moving targets. The question of scheduling feedback from the BS to control feedback for the monitoring period is another open question for investigation.

REFERENCES

[1] J. Kim, P. Bentley, C. Wallenta, M. Ahmed, and S. Hailes, “Danger Is Ubiquitous: Detecting Malicious Activities in Sensor Networks Using the Dendritic Cell Algorithm,” Proc. International Conference on Artificial Immune Systems (ICARIS), pp. 390–403, 2006.

[2] L.N. DeCastro and J. Timmis, "Artificial ImmuneSystems: A New Computational Intelligence Approach," Springer-Verlag, 2002.

[3] U. Aickelin and J. Greensmith, “Sensing Danger: Innate Immunology for Intrusion Detection,” Elsevier Information Security Technical Report, pp. 218–227, 2007.

[4] U. Aickelin, P. Bentley, S. Cayzer, J. Kim, and J. McLeod, “Danger theory: The link between AIS and IDS,” Proceedings of the Second International Conference on Artificial Immune Systems (ICARIS 2003), vol. 2787 of LNCS, Springer-Verlag; pp. 147‑155, 2003.

 ISBN: 978-1-61208-107-6

[5] J. Greensmith, U. Aickelin, and S. Cayzer, “Introducing Dendritic Cells as a Novel Immune-Inspired Algorithm for Anomaly Detection,” Proc. International Conference on Artificial Immune Systems (ICARIS), pp. 153–167, 2005.

[6] N. Mazhar and M. Farooq, “A sense of danger: dendritic cells inspired artificial immune system for MANET security,” GECCO, pp. 63-70, 2008.

[7] D.B. Johnson and D.A. Maltz, “Dynamic source routing in adhoc wireless networks,” T. Imielinski and H. Korth, eds., Mobile computing (Kluwer Academic) chapter5, pp. 153-181, 1996.

[8] J. Greensmith, U. Aickelin, and J. Twycross, “Articulation and Clarification of the Dendritic Cell Algorithm,” ICARIS, LNCS 4163, pp. 404-417, 2006.

[9] A.P. da Silva, M. Martins, B. Rocha, A. Loureiro, L. Ruiz, and H. C. Wong, “Decentralized intrusion detection in wireless sensor networks,” in Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks (Q2SWinet ’05). ACM Press, pp. 16–23, October 2005.

[10] M. Nikraz, G. Caire, and P. A. Bahria, “A methodology for the analysis and design of multi-agent systems using jade,” International Journal of Computer Systems Science and Engineering, vol. 21, no. 2, 2006.

[11] M. Vinyals, J. A. Rodríguez-Aguilar, and J. Cerquides, “A Survey on Sensor Networks from a Multi-Agent perspective,” 2th International Workshop on Agent Technology for Sensor Networks (ATSN), 2008.

[12] R. Biswas, S. Thrun, and L.J. Guibas, “A probabilistic approach to inference with limited information in sensor networks,” in: Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN’04), Berkeley, CA, pp. 269-276, Apr. 2004.

[13] J. Kim and P.J. Bentley, “Evaluating Negative Selection in an Artificial Immune System for Network Intrusion Detection,” Proc. Genetic and Evolutionary Computation Conference (GECCO), pp. 1330-1337, 2001.

[14] S. Forrest, S. Hofmeyr, and A. Somayaji, “Computer Immunology,” Communications of the ACM, 40(10), pp. 88-96, 1997

Study on the Improved D-S Evidence Theory and its Application on Gas Outburst Prediction

Y. Z. Miao[1,2], Y. Chen[1], H. X. Zhang[2] , J. W. Zhang[2],

1. School of Information and Electrical Engineering, China University of Ming and Technology Jiefang South Road, 221008, Xuzhou, China miaoyanzi81@163.com

2. Institute of Technical Aspects of Multimodal Systems, Department of Informatics, University of Hamburg Vogt-Koelln-Strasse 30, 22527, Hamburg, Germany miao@informatik.uni-hamburg.de

***Abstract*-This paper introduces improved combination rules for the D-S Evidence Theory for dealing with the evidence conflicts which considers the coherence evidence and the conflicts evidences together and allocate the conflicts to various focal elements according to the credibility of the coherence evidence. Also after introduce the similarity degree to denote the similarity between two fuzzy focal elements, and extended the *Bel* and *Pl* functions for processing fuzzy data, the improved extension combination rules of the D-S evidence theory to fuzzy sets are described. The coal and gas outburst prediction experiments show that the fusion result with the improved combination rule of the D-S Evidence Theory is more reasonable and could give a more certain decision than each independent method.**

Keywords-D-S Evidence Theory; Gas Outburst Prediction; Fuzzy Sets.

I. Introduction

According to the information supplied by every sensor in a multi-sensor system, information fusion is a reasoning technology for object recognition and property judgment, but information from different sensors is usually abridged, imprecise and even inconsistent. As an uncertain reasoning method and as the generalization of the Bayesian reasoning method, the Dempster-Shafer Evidence Theory (D-S Evidence Theory) is put forward by Dempster in 1967 and expanded by Shafer by systemizing and elaborating the theory. Due to the advantages of the uncertain denotation, measurement and combination, the D-S Evidence Theory was applied widely [1-4]. As it can be combined with other methods, the Evidence Theory is more widely usable and can be extended very well in the future[5-7].

The Evidence Theory has many advantages, but it is not perfect in practical application, and even produces the opposite result to what our intuition tells us. The main reason for these deficits is the variance or conflict between the evidence[8-10]. And the evidence conflict is a problem that cannot be neglected during practical information fusion. The D-S Evidence Theory based on classical sets sometimes seems restrained and helpless concerning the fuzzy concept, so an advanced generalization of extending the evidence theory to fuzzy sets is proposed to overcome the existing insufficiencies.

Since the gas outburst under the coal mine is a very complicated phenomenon, and there are many factors associated with the outburst [11, 12]. Also the warning signs before it happens are also unexpected and changing. So the D-S Evidence Theory which is an intelligent way of uncertain reasoning is the only potent approach to consider multiple associated factors and make a precise prediction.

In this paper, an improved D-S evidence theory is presented, and it is validated in the coal and gas outburst prediction experiments. In Section II, an improved D-S evidence theory is described, which can resolve the evidence conflict problem effectively. In Section III, we propose the generalized combination rule on fuzzy sets by extending the belief function Bel and the plausibility function Pl in the D-S evidence theory to fuzzy sets. And the Section IV introduces the application on outburst prediction with our improved combination rules of the D-S evidence theory on fuzzy sets. In the end, the related experimental results and conclusion would be given in Section V.

II. Improved Evidence Combination Rules

A. *Introduction of Correlative Conceptions*

As proposed in [13] the concept of evidence distance, supposing Θ is the frame of discernment (FOD) including N different propositions and M evidence sources: $S_1, S_2, \ldots, S_M$, the corresponding BPAFs are: $m_1, m_2, \ldots, m_M$. We consider each evidence source S_i as row vector S_i in 2^N dimension; every part of the vector is the distributed probability m_i of every element in exponential sets 2^Θ of Θ. So, the distance between two evidence sources S_i, S_j is defined as:

$$d(S_i, S_j) = \sqrt{\frac{1}{2}(\mathrm{S_i} - \mathrm{S_j})\underline{\underline{D}}(\mathrm{S_i} - \mathrm{S_j})^{\mathrm{T}}} \quad (1)$$

where, $\underline{\underline{D}}$ is a matrix in $2^N \times 2^N$

$$\underline{\underline{D}}(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad A, B \in 2^\Theta$$

 ISBN: 978-1-61208-107-6

The formula to get the evidence distance is also given in [13]:

$$d(S_i, S_j) = \sqrt{\frac{1}{2}\left(\|S_i\|^2 + \|S_j\|^2 - 2\langle S_i, S_j\rangle\right)} \tag{2}$$

in which,

$$\langle S_i, S_j\rangle = \sum_{s=1}^{2^N} \sum_{t=1}^{2^N} m_i(A_s) m_j(A_t) \frac{|A_s \cap A_t|}{|A_s \cup A_t|} \qquad A_s, A_t \in 2^{\Theta}$$

So, the evidence distance represents the difference between two kinds of evidence, which is another reflection of the evidence conflict. On the other hand, the coherence degree of two kinds of evidence is also reflected. Obviously, the distance and coherence between two kinds of evidence are inversely proportional. When the distance between two kinds of evidence is equal to be 1, the two kinds of evidence are absolutely incoherent and the degree of coherence is 0; and when the distance becomes 0, the evidence is absolutely coherent and the degree of coherence is 1. Based on the fusion effect, we choose a simple formula to define the degree of coherence between two kinds of evidence. For example, the degree of coherence between S_i, S_j is described as:

$$coh(S_i, S_j) = 1 - d(S_i, S_j) \tag{3}$$

As can be seen from the definition, the coherence degree reflects the degree of mutual sustainment, so the degree to which evidence S_i is sustained by other instances of evidence is defined as:

$$\sup(S_i) = \sum_{j=1, j\neq i}^{M} coh(S_i, S_j) \tag{4}$$

Hence, the greater the degree to which the evidence is sustained by other evidence, the higher its reliability. Therefore we can define the credibility of a single evidence source using the sustainment degree between two kinds of evidence. The credibility of evidence source S_i is defined as:

$$cre(S_i) = \sup(S_i) \Big/ \sum_{i=1}^{M} \sup(S_i) \tag{5}$$

The more number of highly coherent evidence sources, the more highly credible single evidence sources, and then the higher total credibility of M evidence sources; whereas it is lower. Furthermore, when all the evidence sources conflict absolutely, the coherence degree between two random kinds of evidence is 0 and the total credibility of the evidence sources should intuitively be 0. On the other hand, when all the evidence is absolutely coherent, the mutual coherence degree is 1 and the total credibility of the evidence sources should intuitively be 1. Therefore, we define the total credibility hypersphere of the evidence sources and the total credibility hypersphere of absolute coherence (the perfect hypersphere), the dimensions of which are both C^2_M, and the respective radiuses r and R are defined as:

$$r = \sqrt[C_M^2]{\sum_{i=1}^{M-1} \sum_{j=i+1}^{M} coh^{C_M^2}(S_i, S_j)} \Big/ 2 \tag{6}$$

$$R = \sqrt[C_M^2]{C_M^2 \times 1^{C_M^2}} \Big/ 2 = \sqrt[C_M^2]{C_M^2} \Big/ 2 \tag{7}$$

We consider the total credibility of the evidence sources as the degree closest to the credibility of perfect evidence sources in terms of absolute coherence, that is, the degree of the total credibility hypersphere is closest to the perfect hypersphere of evidence sources. Considering that the credit of evidence sources is one-dimensional, we define the total credit C of evidence sources as:

$$C = r/R \tag{8}$$

B. Improved Combination Rules I

In this paper, we consider the conflict and coherence of the evidence combination synchronously, the conflict evidence combined with the credibility of the evidence source, and the coherence evidence combined with the *AND*-algorithm, which can reflect the degree of intersection fusion. Therefore, our improved rules are respectively based on the two different ideas mentioned above. To this end, we suppose A_i, B_j, and C_k to be the focal elements of M different evidence sources in the following section.

Based on the idea that conflict evidence can be used by parts, we think a part of the conflict information can be allocated to focal elements of the evidence, and the other parts to the unknown term $m(\Theta)$, moreover, the allocation depends on the credibility of the total evidence source and the single evidence. So the improved rule I is:

$$m(\Phi) = 0 \tag{9}$$

$$m(A) = \frac{1-K}{P} \cdot \sum_{A_i \cap B_j \cap C_k \cap \cdots = A} \frac{|A|^M}{|A_i||B_j||C_k|\cdots} \cdot m_1(A_i) m_2(B_j) m_3(C_k)\cdots + KC\sum_{i=1}^{M}\left(m_i(A)\cdot cre(s_i)\right) \qquad \forall A \neq \Phi, \Theta \tag{10}$$

$$m(\Theta) = \frac{1-K}{P} \cdot \sum_{A_i \cap B_j \cap C_k \cap \cdots = \Theta} \frac{|\Theta|^M}{|A_i||B_j||C_k|\cdots} \cdot m_1(A_i) m_2(B_j) m_3(C_k)\cdots + KC\sum_{i=1}^{M}(m_i(\Theta)\cdot cre(S_i)) + K(1-C) \tag{11}$$

Where, the definitions of C and cre(Si) are the same as above. And in which,

$$K = \sum_{A_i \cap B_j \cap C_k \cap \cdots = \Phi} m_1(A_i) \cdot m_2(B_j) \cdot m_3(C_k) \cdot ... \tag{12}$$

This stands for the total conflict between all evidence.

$$P = \sum_{A_i \cap B_j \cap C_k \cap \cdots \neq \Phi} Q \cdot m_1(A_i) \cdot m_2(B_j) \cdot m_3(C_k) \cdot ...$$
$$= \sum_{A_i \cap B_j \cap C_k \cap \cdots \neq \Phi} \frac{|A_i \cap B_j \cap C_k \cap \cdots|^M}{|A_i||B_j||C_k|\cdots} m_1(A_i) \cdot m_2(B_j) \cdot m_3(C_k) \cdot ... \tag{13}$$

The formula above gives the total probability of coherence evidence after intercross fusion; the Q in the formula reflects the degree of intercross fusion of the evidence, including the AND-algorithm. Obviously, the more coherence there is between evidence sources, the higher the degree of intercross fusion; and (1-K)/P stands for the probability of unitary allocation of the coherence evidence after the intercross fusion. The second term in formula (10) and (11) stands for allocating the total conflict of the evidence source by weight logically, respectively considering the credit of the evidence source in total and individually.

Then we prove that function *m* from the rule above is a BPAF, that is, we only need to prove that:

$$\sum_{A \subset \Theta} m(A) = 1 \tag{14}$$

This detailed proof has been given in other paper which has been published [14].

C. Improved Combination Rules II

Based on the idea that conflict evidence can be used completely (in the above rule) under the condition that the total credibility of the evidence sources *C=1*, and referring to the method in reference [15], we think the conflict information can be completely allocated to focal elements of evidence, and the allocation depends on the credibility of every single evidence source. Therefore, the improved rule *II* is as follows:

$$m(\Phi) = 0 \tag{15}$$

$$m(A) = \frac{1-K}{P} \sum_{A_i \cap B_j \cap C_k \cap \cdots = A} \frac{|A|^M}{|A_i||B_j||C_k|\cdots} m_1(A_i) m_2(B_j) m_3(C_k) \cdots + K \sum_{i=1}^{M} \left(m_i(A) \cdot cre(s_i) \right) \quad \forall A \neq \Phi \tag{16}$$

Every term in the above formula is the same as that in rule *I*. When the frame of discernment Θ is also the focal element, we can calculate the probability *m (Θ)* using formula (16).

According to rule *II*, we also have the summary of the probability ∑*m(A)* is equal to be 1, and the reasoning is similar to that in rule I, except that the total credibility of the evidence source used is *C=1*. So the m is also a BPAF function in rule *II*.

From the above proving process, we can conclude that rule *II* can be regarded as an example for rule *I* if we consider the total credit of the evidence source in rule I to be *C=1*, that means, the conflict information can be allocated to focal elements of evidence and unknown terms according to the total credibility of the evidence source in rule *I*. Therefore, rule *I* is more reliable and conservative than rule *II*, while rule *II* is a more changeable and less reliable decision-making method. However, in contrast to rule *I*, it is flexible and can adapt to special requirements.

III. THE IMPROVED EXTENTION OF FUZZY D-S EVIDENCE THEORY

As we know, the evidence theory can express the “uncertainty” distinctly and correctly, and it also features the D-S combination rule which is based on sound mathematic rules. As the evidence theory has a good effect of combination, many researchers continually improved the D-S formulas. However, by extending it from classic sets to fuzzy sets, the denotation and algorithm of fuzzy sets and their intrinsic meanings underwent great changes compared to classic sets. Therefore, when we extend the evidence theory to fuzzy sets, we should change the way evidence is denoted as well as the corresponding combination rule.

Classical sets only need to consider the included elements which are certain, but fuzzy sets consider not only the included elements, but also the degree to which every element is subjected to the fuzzy sets. So taking the fuzzy sets into account, we cannot only heed the included elements like the classic sets, as there several degrees to which an element “belongs to” the set. These degrees are confirmed only by the subjection degree.

Example1: Three fuzzy sets:

$$\widetilde{A} = \{\frac{0.9}{1}, \frac{1}{2}, \frac{0.9}{3}, \frac{0.8}{4}, \frac{0.6}{5}, \frac{0.3}{6}, \frac{0.1}{7}\},$$
$$\widetilde{B} = \{\frac{0.1}{2}, \frac{0.1}{3}, \frac{0.1}{4}, \frac{0.2}{5}, \frac{0.5}{6}, \frac{0.8}{7}, \frac{1}{8}, \frac{0.9}{9}\},$$
$$\widetilde{C} = \{\frac{0.9}{2}, \frac{1}{3}, \frac{0.9}{4}, \frac{0.8}{5}, \frac{0.4}{6}, \frac{0.2}{7}, \frac{0.1}{8}, \frac{0.1}{9}\},$$

According to the definition of the intersection of fuzzy sets [16], we have:

$$\widetilde{A} \cap \widetilde{B} = \{\frac{0.1}{2}, \frac{0.1}{3}, \frac{0.1}{4}, \frac{0.2}{5}, \frac{0.3}{6}, \frac{0.1}{7}\}$$
$$\widetilde{A} \cap \widetilde{C} = \{\frac{0.9}{2}, \frac{0.9}{3}, \frac{0.8}{4}, \frac{0.6}{5}, \frac{0.3}{6}, \frac{0.1}{7}\}$$

From the formulas above we see that the intersections $\widetilde{A} \cap \widetilde{B}$ and $\widetilde{A} \cap \widetilde{C}$ seem to have the “same” fuzzy elements on the surface, but a different subjection degree. But according to the definition of fuzzy sets, the two fuzzy sets have a very different nature. Therefore, when considering fuzzy sets, the subjection degree of every element to the fuzzy set is more important, and it is necessary to extend the combination rules of the evidence theory to fuzzy sets.

Example2: Supposing the frame of discernment is $\Theta = \{\theta_1, \theta_2, \cdots, \theta_n\}$, and the fuzzy set is $\tilde{A} = \{\frac{0.9}{\theta_1}, \frac{1}{\theta_2}, \frac{0.9}{\theta_3}, \frac{0.8}{\theta_4}, \frac{0.6}{\theta_5}\}$, the degree of subjection of the elements $\theta_i\,(i = 6,7,\cdots,n)$ to the fuzzy set $\tilde{A}$ is equal to 0. The subjection degree of every element in the frame of discernment Θ to an empty set also seems to be 0, so the subjection degrees of elements $\theta_i(i = 6,7,\cdots,n)$ to an empty set and a fuzzy set $\tilde{A}$ are both equal to 0. This is the similarity between them, thus the degree of their similarity should be larger than 0. This means that the frame of discernment can be considered as a special fuzzy set with subjection degrees of equal to 1 for all included elements, therefore the empty set and the frame of discernment Θ can be considered as absolute opposites, and the degree of their similarity should be 0. Furthermore, the similarity of any fuzzy set to itself should be equal to 1.

According to the definitions of the contribution factor, the belief function and the similarity function of fuzzy evidence reasoning were described as follows:

$$Bel(\tilde{B}) = \sum_i F_*(\tilde{B}, \tilde{A}_i) m(\tilde{A}_i) \tag{17}$$

$$Pl(\tilde{B}) = \sum_i F^*(\tilde{B}, \tilde{A}_i) m(\tilde{A}_i) \tag{18}$$

Taking the important degree of every element in the frame of discernment into account, the following formulas of the corresponding contribution factor result:

$$F_*(\tilde{B}, \tilde{A})_\omega = 1 - \frac{1}{|\tilde{A}|} \sum_i^{|\tilde{A}|} \omega_i \left| \mu_{\tilde{B}}(\theta_i) - \mu_{\tilde{A}}(\theta_i) \right| \tag{19}$$

$$F^*(\tilde{B}, \tilde{A})_\omega = 1 - \frac{1}{|\Theta|} \sum_i^{|\Theta|} \omega_i \left| \mu_{\tilde{B}}(\theta_i) - \mu_{\tilde{A}}(\theta_i) \right| \tag{20}$$

Putting the formulas of $F_*(\tilde{B}, \tilde{A})_\omega$ and $F^*(\tilde{B}, \tilde{A})_\omega$ above into $F_*(\tilde{B}, \tilde{A}_i)$ and $F^*(\tilde{B}, \tilde{A}_i)$ in the formulas (17) and (18) separately, we get a measurement method for the weight of the belief function and the similarity function.

The combination rule of fuzzy evidence reasoning adopted the idea of Haenni[17], which is to modify the belief allocation model, not to change the form of Dempster's combination rule which has well characters. Before combining the evidence, the Basic Probability of Assignment Function (BPAF) of the fuzzy focal elements needs to be amended. Based on the similarity between fuzzy sets, for amending the BPAF of fuzzy focal element $\tilde{A}$, the weight of the similarity between fuzzy focal elements $\tilde{C}$ and $\tilde{A}$ was confirmed as:

$$\omega(\tilde{C}, \tilde{A}) = 1 - \frac{1}{|\Theta|} \sum_i \left| \mu_{\tilde{C}}(\theta_i) - \mu_{\tilde{A}}(\theta_i) \right| \tag{21}$$

Supposing Bel_1 and Bel_2 are the belief functions of the same frame of discernment $\Theta = \{\theta_1, \theta_2, \cdots, \theta_n\}$, it has the basic probability assignment function m_1 and m_2, the fuzzy focal elements of which are $\{\tilde{A}_1, \tilde{A}_2, \cdots, \tilde{A}_p\}$ and $\{\tilde{B}_1, \tilde{B}_2, \cdots, \tilde{B}_q\}$, so the BPAF $m : 2^\Theta \to [0,1]$ of a nonempty set $\tilde{C}$ can be put forward as:

$$m(\tilde{C}) = m_1 \oplus m_2(\tilde{C}) = \frac{\sum_{\tilde{A}_i \cap \tilde{B}_j = \tilde{C}} \omega(\tilde{C}, \tilde{A}_i) m_1(\tilde{A}_i) \omega(\tilde{C}, \tilde{B}_j) m_2(\tilde{B}_j)}{1 - \sum_{\tilde{A}_i, \tilde{B}_j} (1 - \omega(\tilde{A}_i \cap \tilde{B}_j, \tilde{A}_i) \omega(\tilde{A}_i \cap \tilde{B}_j, \tilde{B}_j)) m_1(\tilde{A}_i) m_2(\tilde{B}_j)} \tag{22}$$

in which, $\omega(\tilde{C}, \tilde{A}_i)$ is the weight of the fuzzy focal element $\tilde{A}_i (i = 1,2,\cdots,p)$, $\omega(\tilde{C}, \tilde{B}_j)$ is the weight of the fuzzy focal element $\tilde{B}_j (j = 1,2,\cdots,q)$.

Accordingly, taking the important degree of the elements in the frame of discernment $\Theta = \{\theta_1, \theta_2, \cdots, \theta_n\}$ into account, the corresponding weight and combination rule could be described as:

$$\omega'(\tilde{C}, \tilde{A}) = 1 - \frac{1}{|\Theta|} \sum_i \alpha_i \left| \mu_{\tilde{C}}(\theta_i) - \mu_{\tilde{A}}(\theta_i) \right| \tag{23}$$

$$m'(\tilde{C}) = m_1 \oplus m_2'(\tilde{C}) = \frac{\sum_{\tilde{A}_i \cap \tilde{B}_j = \tilde{C}} \omega'(\tilde{C}, \tilde{A}_i) m_1(\tilde{A}_i) \omega'(\tilde{C}, \tilde{B}_j) m_2(\tilde{B}_j)}{1 - \sum_{\tilde{A}_i, \tilde{B}_j} (1 - \omega'(\tilde{A}_i \cap \tilde{B}_j, \tilde{A}_i) \omega'(\tilde{A}_i \cap \tilde{B}_j, \tilde{B}_j)) m_1(\tilde{A}_i) m_2(\tilde{B}_j)} \tag{24}$$

where α_i is the weight of $\theta_i \in \Theta$, $\omega'(\tilde{C}, \tilde{A}_i)$ is the weight of fuzzy focal element $\tilde{A}_i (i = 1,2,\cdots,p)$, $\omega'(\tilde{C}, \tilde{B}_j)$ is the weight of fuzzy focal element $\tilde{B}_j (j = 1,2,\cdots,q)$.

This above idea of the combination rule put forward by *Li*n Zhigui in [18] seems more reasonable, and using numeral experiments, he validated the advantage of the defined concept of similarity between fuzzy sets and the combination rule, which is more sensitive to the changing of fuzzy focal elements. However, this concept also has deficiencies. Firstly, he only defined the similarity between fuzzy focal elements, and according to the combination formula, the similarity degree is equal to 0 between an empty set and any nonempty fuzzy set. Secondly, the combination formula he put forward has a deficiency, that is, the function m after combination is no longer a basic probability assignment function, because now $\sum m(\widetilde{C}) < 1$, and it cannot satisfy the basic condition $\sum m(\widetilde{C}) = 1$.

From the above chain of reasoning, we can define the similarity between random fuzzy sets as follows:

Definition 1 Supposing the frame of discernment to be $\Theta = \{\theta_1, \theta_2, \cdots, \theta_n\}$, and the fuzzy sets $\widetilde{A}$ and $\widetilde{C}$ are the two random fuzzy subsets, then the similarity between fuzzy sets $\widetilde{A}$ and $\widetilde{C}$ is defined as:

$$\omega(\widetilde{C}, \widetilde{A}) = 1 - \frac{1}{|\Theta|} \sum_i \left| \mu_{\widetilde{C}}(\theta_i) - \mu_{\widetilde{A}}(\theta_i) \right| \tag{25}$$

As this definition of real function ω satisfies the four basic conditions in reference [17], the definition of the similarity above is reasonable. And based on the property of the basic probability assignment function, we define the combination rule of the evidence theory on fuzzy sets as definition 2.

Definition 2 Supposing Bel_1 and Bel_2 to be the belief functions of the same frame of discernment $\Theta = \{\theta_1, \theta_2, \cdots, \theta_n\}$, the basic probability assignment function of which to be m_1 and m_2, and the fuzzy focal elements to be $\{\widetilde{A}_1, \widetilde{A}_2, \cdots, \widetilde{A}_p\}$ and $\{\widetilde{B}_1, \widetilde{B}_2, \cdots, \widetilde{B}_q\}$, then the basic probability assignment function is defined to be $m: 2^\Theta \to [0,1]$:

$$m(\varnothing) = 0 \tag{26}$$

$$m(\widetilde{C}) = m_1 \oplus m_2(\widetilde{C}) \tag{27}$$

$$= \frac{\sum_{\widetilde{A}_i \cap \widetilde{B}_j = \widetilde{C}} \omega(\widetilde{C}, \widetilde{A}_i) m_1(\widetilde{A}_i) \omega(\widetilde{C}, \widetilde{B}_j) m_2(\widetilde{B}_j)}{\sum_{\widetilde{A}_i} \sum_{\widetilde{B}_j} \omega(\widetilde{A}_i \cap \widetilde{B}_j, \widetilde{A}_i) \omega(\widetilde{A}_i \cap \widetilde{B}_j, \widetilde{B}_j) m_1(\widetilde{A}_i) m_2(\widetilde{B}_j) - \sum_{\widetilde{A}} \sum_{\widetilde{B}_j, \widetilde{A}_i \cap \widetilde{B}_j = \varnothing} \omega(\varnothing, \widetilde{A}_i) \omega(\varnothing, \widetilde{B}_j) m_1(\widetilde{A}_i) m_2(\widetilde{B}_j)}$$

Then we also prove that the function from the formulas (26) and (27) above is a BPAF, that is, we only need to prove that $\sum_{\widetilde{C}} m(\widetilde{C}) = 1$ in other reference [19].

IV. Application on Gas Out Burst Prediction

In order to validate the improved combination rule of the D-S evidence theory on fuzzy sets, we did our experiments of gas outburst prediction with this method. The intelligent prediction way based on multi-sensor information presented in this paper is choosing the preliminary prediction results from different main methods as the evidences of the D-S evidence theory, and then making an extended reasoning fusion decision with the improved fuzzy combination rule.

The data in Table I show the indexes from five different mines and the actual results are divided into two kinds, The "A" stands for occurrence of outburst, "B" is no occurrence.

TABLE I The Indexes from Different Areas

Number	Preliminary Velocity	Solid Coefficient f	Gas Pressure *P (Mpa)*	S_{max} (L/m)	Composite Index K	Composite Index D	Result
19	10.5	0.3297	0.97	11.2	32.02	3.43	A
20	12	0.2252	0.40	5.3	53.29	−7.89	A
21	3	0.4762	0.21	4.8	6.30	−8.29	B
22	9.1	0.4167	0.63	5.5	14.4	−2.00	A
23	4	0.6610	0.47	3.9	6.05	−2.40	B

In this experiment, we chose neural network, single index method--Preliminary Velocity, D-K composite index, and cutting desorption index $S_{\max}$ way as four evidence source. The prediction results from these four ways are gotten respectively not only for comparison, but also for evidence source of extended decision of the D-S Evidence Theory.

To compare the fusion result with each of the single methods above, we achieved the following prediction results listed in Table II as follows:

TABLE II Fusion Results of Each Method

No.		19	20	21	22	23
Neural Network	$m_1(A)$	0.9133	1	0.0133	0.9998	0.1181
	$m_1(B)$	0.0867	0	0.9867	0.0002	0.8819
Single Index	$m_2(A)$	1	1	0.0926	0.8606	0.1220
	$m_2(B)$	0	0	0.9074	0.1394	0.8780
Composite Index	$m_3(A)$	1	1	0.0620	0.9328	0.0588
	$m_3(B)$	0	0	0.9380	0.0672	0.9412
Cutting Desorption	$m_4(A)$	1	0.9524	0.3571	1	0.0816
	$m_4(B)$	0	0.0476	0.6429	0	0.9184
D-S Fusion	$m(A)$	0.9982	0.9994	0.0554	0.99	0.0316
	$m(B)$	0.0018	0.0006	0.9446	0.01	0.9684

 ISBN: 978-1-61208-107-6

From this fusion results, we can make a determinate decision, areas in 19, 20, 22 have the risk of outburst, and areas in 21, 23 have no risk of outburst, which accord with the factual results.

From the table above, we can conclude that the result after fusion with D-S evidence theory is more reasonable than the result from each single present prediction method. Also it is much easier to make a determinate decision from the fusion result. The results proved that the evidence theory fusion could "compensate" the deficiencies in each single prediction, thus adding to other evidence sources. For example, according to the data from area in 21, if we only use cutting desorption index method, the prediction result is uncertain, the probability of no risk is only 64.29%, and we could not give an assured conclusion, but after "compensating" of other methods, the result is improved, the probability of no risk is up to 94.46%, and we can give the conclusion exactly. Another example, the prediction in area 23, the credibility of no risk after evidence fusion is up to 96.84%, which is higher than the credibility from each other method.

V. CONCLUSION

This paper improves the combination rules of the D-S Evidence Theory to deal with coherent or incoherent evidence obtained from multiple sources. According to the credibility of coherence evidence, the improved rules allocate the conflicts to various focal elements, so the new rules can process both highly conflicting and coherent evidence effectively, and the rules can provide reasonable results with better convergence efficiency than other rules in the case of highly conflicting evidence sources.

We introduced the similarity degree to denote the similarity between two fuzzy focal elements, and extended the *Bel* and *Pl* functions for processing fuzzy data. Then the improved extension combination rule of the D-S evidence theory to fuzzy sets is considered.

The coal and gas outburst prediction experiments show that the fusion result with the improved combination rule of the D-S Evidence Theory is more reasonable and could give a more certain decision than each independent method.

In the future work, the experience of the experts could be added as a new evidence source, combing external measurement data with human subjective experience, the fusion result must be better and more reasonable.

REFERENCES

[1] X. Hu and C. Xie,"Security Operation Center Design Based on D-S Evidence Theory", *Mechatronics and Automation, Proceedings of the 2006 IEEE International Conference ,* Luoyang, China, pp.2302 -2306, June ,2006.

[2] H. Su and G. Zheng, "A Non-Intrusive Drowsiness Related Accident Prediction Model Based on D-S Evidence Theory", *Proceedings of the 1st International Conference on Bioinformatics and Biomedical Engineering, Wuhan, China,* pp.570 -573, July, 2007.

[3] M. Bauer. "Approximations for decision making in the Dempster-Shafer Theory of evidence", *Proceedings of the Twelfth annual conference on Uncertainty in Artificial Intelligence (UAI-96)*, San Francisco, California. Morgan Kaufmann, pp. 73-80, 1996.

[4] V. Kaftandjian, Y. Zhu, O. Dupuis, and D. Babot, "The combined use of the evidence theory and fuzzy logic for improving multimodal nondestructive testing systems", *IEEE Transactions on Instrumentation and Measurement*, vol.54, no.5, pp.1968-1977, 2005.

[5] F. Rottensteiner, J. Trinder, and S. Clode, "Using the Dempster-Shafer method for the fusion of LIDAR data and multi-spectral images for building detection. Information fusion", *Proceedings of the 7th International Conference on Information Fusion*, pp. 1396-1403, 2005.

[6] S. Regis, J. Desachy, and A. Doncescu, "Evaluation of biochemical sources pertinence in classification of cell's physiological states by evidence theory", *Proceedings of 2004 IEEE International Conference on Fuzzy Systems*, pp. 867-872, 2004.

[7] D. Tan, X. Yan, S. Gao , and Z. Liu, "Fault diagnosis for spark ignition engine based on multi-sensor data fusion", *Proceedings of IEEE International Conference on Vehicular Electronics and Safety*, pp..311-314, 2005.

[8] Y. Xu, X. Wu, J. Shen, and X. San, "Feature fusion algorithm based on DT-CWT texture*". 7th International Conference on Signal Processing, (ICSP '04),* vol.2, pp.1492-1495, 2004.

[9] W. Lu, Q. Wu, and Q. Shao, "The D-S Evidence Theory and Application to Feasibility Evaluating of New Product Developing", *Journal of Operational Research and Management,* vol.13,no.5, pp.111-115, 2004.

[10] L. Xu, Z. Lin, and S. Yang, "Vague Sets Based Evidence Combinational Rule With Generalized Belief Function", *Proceedings of IEEE International Conference on Information Acquisition,* pp.764-769, 2006.

[11] R. Zhang. "Application of advanced information technology on coal and gas outburst prediction" *Ph.D thesis, Chongqing University*, 2004.

[12] C. Li. "Research on gay classifiction and prediction for coal and gas outburst quicksand", *Ph.D thesis, China University of Mining and Technology,* 2005.

[13] A. Jousselme, D. Grenier, and E. Bosse, "A new distance between two bodies of evidence", Journal of Information Fusion", vol. 3, pp. 237-239, 2002.

[14] Y. Miao, J. Zhang, H. Zhang, and X. Ma. Improvement of the Combination Rules of the D-S Evidence Theory Based on Dealing with the Evidence Conflict. Proceedings of the *2008 IEEE International Conference on Information and Automation ,* pp..331-336, 2008.

[15] L. Hegarat, S. Mascle, D. Richard, and C. Ottle "Multi-scale data fusion using Dempster-Shafer evidence theory", *Proceedings of IEEE International Conference on Geoscience and Remote Sensing Symposium*, vol.2, pp. 911-913, 2002.

[16] S. Chen, J. Li, and X. Wang,,"Fuzzy set theory and its application", Beijing, Science publish company, 2005.

[17] R. Haenni, "Are alternatives to Dempster's rule of combination real alternatives? Comments on "About the belief function combination and the conflict management problem"". *Information Fusion*, Vol.3(4), pp.237-239, 2002.

[18] Z. Lin, "Study on Information Fusion Based on Evidence Theory and its Application in Water Quality Monitoring". *Ph.D. thesis, Nanjing Hohai University,* 2005.

[19] Y. Miao, X. Ma, H. Zhang, and J. Zhang " An Improved Extension of the D-S Evidence Theory to Fuzzy sets". *Proceedings of the 3th International Multi-Conference on Computing in the Global Information Technology, Athens, Greece,* pp.148-153, 2008.

An Optimal Resource Assignment Problem in Smart Grid

Prakash Ranganathan
Department of Electrical Engineering
University of North Dakota
Grand Forks, ND, USA
e-mail:prakashranganathan@mail.und.edu

Kendall Nygard
Department of Computer Science
North Dakota State University
Fargo, ND, USA
e-mail:Kendall.Nygard@ndsu.edu

Abstract— **The paper describes a resource allocation problem in a smartgrid application formulated and solved as a binary integer programming model. For handling power outages from the main distribution circuit, the intelligent agents in the smart grid have to utilize and negotiate with DER (distributed energy resource) agents that act on behalf of the local generators in the grid, to negotiate power supply purchases to satisfy shortages. We develop a model that can assign these DERs optimally to available multiple regional utility areas or units (RUAs) that are experiencing power shortages. This is a resource assignment problem. The DERs in our model depict the behavior of power generated through windturbine, solar powered generation or other renewable power generation units and the region or area refers to a centralized distribution unit. The integer programming approach is called a capacity based Iterative Binary Integer Linear Programming (C-IBILP). All simulation results are carried out using the optimization tool box in MATLAB. Computation results exhibits very good performance for problem instances tested and validates the assumptions made.**

***Keywords*— C-IBILP; DER; RUA; BB.**

I. INTRODUCTION

Dynamic real-time power systems often operate in continuously changing environments such as adverse weather conditions, sudden transformer failures, malfunctioning of a sub-system of a transmission or distribution network. These disruptions along with the complexity of our power network systems, cause the energy demand and loads of a power system to fluctuate, potentially resulting in widespread outages and huge price spikes. Data from the North American Electric Reliability Council (NERC) and analyses from the Electric Power Research Institute (EPRI) indicate that average outages from 1984 to the present time have affected nearly 700,000 customers per event annually [1]. Smaller outages occur much more frequently and affect tens to hundreds of thousands of customers every few weeks or months, while larger outages occur every two to nine years and affect millions. Although preventing these outages remain challenging, such changes (increases or decreases) in demand by consumers can often be offset by distributed energy resources (DERs), which are renewable resources like solar and wind based power to satisfy the shortages or reduce the outage levels. In our work, we consider the use of such DER-based standby mechanisms and formulate to support their issue. We apply an Iterative Binary Integer Linear Programming (IBILP) technique [2] to optimally assign DERs to a region based on criteria such as power levels, demands and preferences. A resource allocation for complex power system is robust with respect to variations in demand and fluctuations in power levels. The amount of additional power that DERs can generate and be effectively utilized in power network is a measure of robustness. Hence, we argue that a capacity based the Iterative Binary Integer Linear Programming (C-IBILP) model is inherently a robust resource allocation.

The structure of the remaining paper is as follows: In Section II, an overview and related work for the smart grid is discussed. In Section III, we present a general formulation of this DER assignment problem. In Section IV, we describe how to solve this problem optimally by using a branch-and-bound based (BB) algorithm with equality and inequality constraints. In Section V, we show the experimental results.

II. RELATED WORK

Mathematical programming has enjoyed a burgeoning presence in theoretical computer science, both as a framework for developing algorithms and, increasingly, as a bonafide model of computation whose limits are expressed in terms of sizes of formulations and integrality gaps of formulations [3, 4, 17]. Linear formulations are an appealing model of computation because both optimization and decision problems fit naturally into the framework, and both theoretically tractable and efficient practical algorithms exist for solving linear programs. For instance, state-of-the-art approaches to exactly solving large-scale instances of many NP hard problems rely on integer programming approaches that require the repeated solution of integer programs representing the problems [5]. The polynomial-time algorithms of [15] and other algorithms [13-16] cannot be applicable in this application due to high complexity and extensive run-times. Modification will be investigated in future work. We refer to a fundamental model for DER assignment as the Capacity based Iterative Binary Integer Linear Programming (C-IBILP) model. There has been little attention given to this type of approach in smart electrical grid analyses. To our knowledge, smart-grid problems of this type have not been solved for DER allocations using optimization models that perform optimal matching of

 ISBN: 978-1-61208-107-6

supply sources demand sites using prediction of generation and market-controlled consumption. Such optimization algorithms are comparable to hard unsolved problems in inference, optimization, and control [12].

III. DER-ASSIGNMENT PROBLEM

To illustrate our problem formulation, we assume that there are 7 areas (RUAs) and 6 DER units with the demand and preference levels shown in Fig 1. We define a Regional Utility Area (RUA) as the local distribution power utilities within the micro grid that distributes the power within their network for its loads [1]. For simplicity we name them Area 1, Area 2….Area 7 as illustrated in Figure 1. The power demand and the preferences in Fig.1 depict a demand driven DER assignment problem that also accommodates preference information. The parameters in the figure are for illustration purposes.

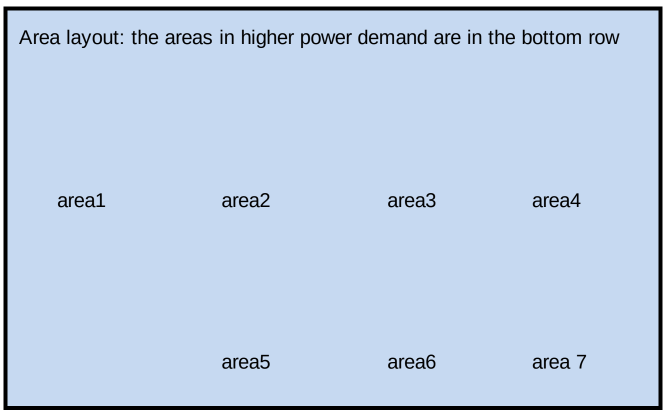


Figure 1. RUA layout

A simple allocation 'text' script on MATLAB would be as follows: text (0.1, .73, 'area1'); text (.35, .73, 'area2'); text (.60, .73, 'area3'); text (.82, .73, 'area4'); text (.35, .42, 'area5'); text (.60, .42, 'area6'); text (.82, .42, 'area 7').

For example, suppose our simulation study is charged with a need to optimally assign 6 DERs, DER1, DER 2, DER3, DER 4, DER 5, DER 6, to 7 regional utility areas (RUA) based on criteria such as capacity in power levels that these DER are able to generate and preferences in the area that these DER wish to operate. For simplicity in our optimization procedure, we also assume that each RUA can have no more than one DER, and each DER gets exactly one RUA. The DER can have a preference for the area that they wish to join, and their preferences are considered based on their capacity, i.e., the more power they have been able to generate the power (kW), the higher the capacity.

We weigh the preferences based on capacity power level of DER'S through a preference weight matrix (pwm), so that the more power that the DER's can generate, the more their preferences count.

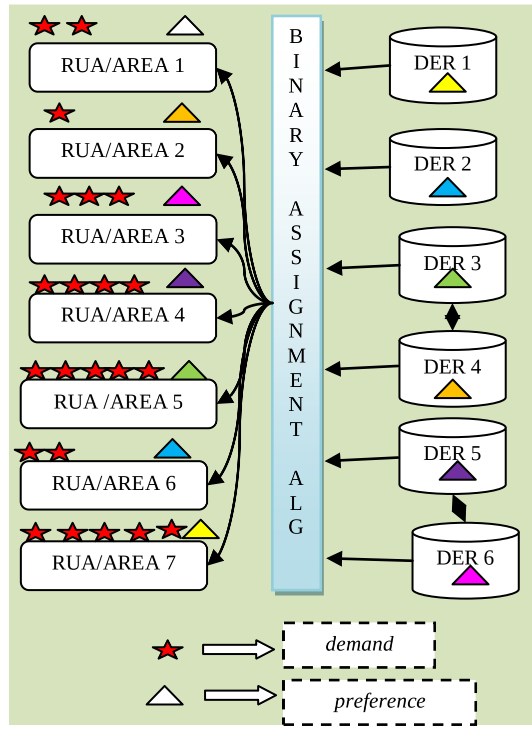


Figure 2. DER vs RUA assignment problem

Also, we impose multiple constraints such as some RUA have demand, some do not, and some demands are higher than others; DER 3 and DER 4 often work together, so we would like them to be no more than one RUA away, and DER 5 and DER 6 often work together so they also should be no more than one RUA away. Our approach to solve the assignment problem is to formulate it as a capacity based Iterative Binary Integer Linear Programming (C-IBILP) model and relax the integrality constraints. Our overall objective is to maximize the satisfaction of the preferences weighted by capacity which will allocate these DERs to their areas. This is done through a binary integer programming model by defining a linear objective function. Our algorithm uses a branch-and-bound procedure with linear programming bounds with 'minimum integer infeasibility' as the branch strategy and 'depth first search for the node search strategy.

To develop our problem formulation, the first step is to choose what each element of our solution vector |x| represents. We use binary integer variables which represents the specific assignments of DERS to RUAs. If the DER is assigned to a RUA, the variable takes the value 1 and if not assigned, the variable takes the value 0. We consider the DER's in sequential order as DER 1, DER 2, DER 3, DER 4, DER 5, DER 6 and DER 7. The nth sequence of elements in vector |x| stores the assignment variables for DER n. In

 ISBN: 978-1-61208-107-6

our example |x (1)| to |x (7)| correspond to DER1 being assigned to Area 1, Area 2, etc., up to Area 7. In all, our vector |x| has 6 sequences of 7 elements each or 42 elements in all. Each sequence has a single binary variable set to 1, enforcing a multiple choice condition for each DER.

A. DER Capacities

We impose constraints based upon DER preference level in their area of operation driven by their capability to generate power. The concept is that the more power that a DER can generate, the higher preference level. For example consider the randomly set power levels given in kiloWatts (kW) below.

a. DER 1→9 kW
b. DER 2→10 kW
c. DER 3→5 kW
d. DER 4→3 kW
e. DER 5→1.5 kW and
f. DER 6→ 2 kW

We create a normalized weight vector based on capacity and also assume that certain DERs should be used in some preferred region or area, such as a DER with higher power generation capability being be used in large demand areas. This normalized weigh vector can be obtained on MATLAB as follows:

capacity = [9 10 5 3 1.5 2];
weight vector = capacity/sum (capacity);

```
>> capacity

capacity =

    9.0000   10.0000    5.0000    3.0000    1.5000    2.0000

>> weightvector

weightvector =

    0.2951    0.3279    0.1639    0.0984    0.0492    0.0656
```

B. RUA Preferences

We set up a preference weight matrix (pwm or prefmatrix) where the rows correspond to AREAS and the columns correspond to DERS. We assume that each DER will give values for each area so that the sum of all their choices, (i.e., their columns), sums to 100. A higher number means the DER prefers the area. We justify the use of the preference matrix by noting that limitations in algorithm scalability and data availability preclude a fully centralized solution to the problem of interest. Thus, decision making must be decentralized, and we accordingly divide the power network into many smaller RUAs, where the prefmatrix concept is applied to individual regions in the network.
An example of DER preferences is shown below:

DER1 = [0; 0; 0; 0; 10; 40; 50];
DER2 = [0; 0; 0; 0; 20; 40; 40];
DER3 = [0; 0; 0; 0; 30; 40; 30];
DER4 = [1; 3; 3; 3; 10; 40; 40];
DER5 = [3; 4; 1; 2; 10; 40; 40];
DER6 = [10; 10; 10; 10; 20; 20; 20];

The i^{th} element of a DER's preference vector is the value the i^{th} RUA. Thus, the combined preference matrix is expressed as 'prefmatrix':

prefmatrix = [DER1 DER2 DER3 DER4 DER5 DER6];

```
>> prefmatrix

prefmatrix =

     0     0     0     1     3    10
     0     0     0     3     4    10
     0     0     0     3     1    10
     0     0     0     3     2    10
    10    20    30    10    10    20
    40    40    40    40    40    20
    50    40    30    40    40    20
```

Case 1: We treat the above 'prefmatrix' arrangement as case 1 for analysis. We then weigh the preferences matrix by the |weightvector| to scale the columns by capacity. We also reshape this matrix as a vector in column-order so that it corresponds to our |x| vector. This is achieved in MATLAB script as follows:

PM = prefmatrix * diag (weightvector);

```
>> diag(weightvector)

ans =

    0.2951         0         0         0         0         0
         0    0.3279         0         0         0         0
         0         0    0.1639         0         0         0
         0         0         0    0.0984         0         0
         0         0         0         0    0.0492         0
         0         0         0         0         0    0.0656
```

c = PM (:);

```
>> PM

PM =

         0         0         0    0.0984    0.1475    0.6557
         0         0         0    0.2951    0.1967    0.6557
         0         0         0    0.2951    0.0492    0.6557
         0         0         0    0.2951    0.0984    0.6557
    2.9508    6.5574    4.9180    0.9836    0.4918    1.3115
   11.8033   13.1148    6.5574    3.9344    1.9672    1.3115
   14.7541   13.1148    4.9180    3.9344    1.9672    1.3115
```

 ISBN: 978-1-61208-107-6

IV. OBJECTIVE FUNCTION AND ADDED CONSTRAINTS

A. Objective function

Our objective is to maximize the total preferences measure weighted by capacity. This is a linear objective function max c'*x or equivalently min -c'*x with c being preferences of DER. We use the BINTPROG script [2] of MATLAB to run our model that is defined as:

$$\min_{x} f^{T}x : \begin{cases} A.x \leq b, \\ Aeq.x = beq, \\ x : \text{binary} \end{cases}$$

where.,

f:Vector containing the coefficients of the linear objective function.
A: Matrix containing the coefficients of the linear inequality constraints $A{\cdot}x \leq b$.
b: Vector corresponding to the right-hand side of the linear inequality constraints.
Aeq: Matrix containing the coefficients of the linear equality constraints $Aeq{\cdot}x = beq$.
beq: Vector containing the constants of the linear equality constraints.
*x0:*Initial point for the algorithm.
Options: Options structure containing options for the algorithm.
x: a binary integer solution vector—that is, its entries can only take on the values 0 or 1.

B. Constraints

The first set of constraints requires that each DER is assigned to exactly one area. For example, since DER2 is the second DER, we enforce the condition that |sum(x (8:14)) =1|. We represent these linear constraints in an equality matrix Aeq and right hand side vector beq, where |Aeq*x = beq|, by building the appropriate matrices. The matrix |Aeq| consists of ones and zeros. For example, the second row of |Aeq| corresponds to DER2 getting exactly one RUA, so the row pattern is the following:

0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 ... 0 0 0

These conditions are implemented in MATLAB code as follows:

```
|Aeq (2, :)*x = 1| is equivalent to |sum(x (8:14)) = 1|.
numAREAS = 7;
numDERS = 6;
numDim = numAREAS * numDERS;
onesvector = ones (1, numAREAS);
Each row of Aeq corresponds to one DER.
Aeq =blkdiag (onesvector, onesvector, onesvector, onesvector, onesvector, onesvector);
beq = ones (numDERS, 1);
view the structure of Aeq, that is, where there are nonzeros (ones) figure;
```

The second sets of constraints are inequalities. These constraints specify that each AREA has no more than one DER in it, i.e., each AREA has one DER in it, or is empty. We build the matrix |A| and the vector |b| such that |A*x <= b| to capture these constraints. Each row of |A| and |b| corresponds to a RUA and so row 1 corresponds to the DER assigned to RUA 1. In this case, the rows have the type of pattern shown below for row 1:

1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 ... 1 0 0 0 0 0 0

Each subsequent row is similar but is shifted (circularly) to the right by one spot by one position. For example, row 3 corresponds to RUA 3 and enforces that |A(3,:)*x <= 1|, so that AREA 3 cannot have more than one DER. Figures 3 and 4 illustrate equality and inequality constraints that are explained above.

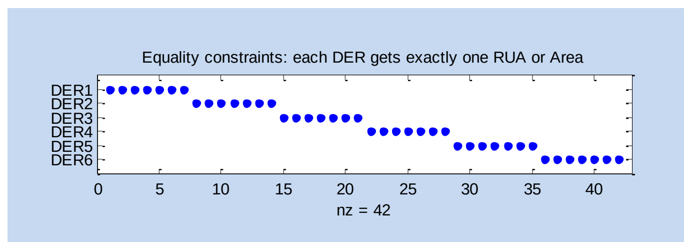


Figure 3. Equality Constraints

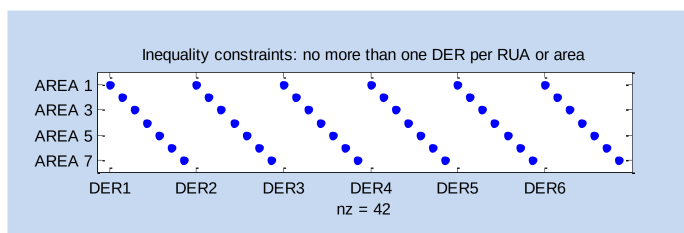


Figure 4. Inequality constraints

```
A = repmat(eye(numAREAS),1,numDERS);
b = ones(numAREAS,1);
```

where '*repmat*' represents the replicate and tile array. The elements of next set of constraints are also inequalities, so they are added to the matrix |A| and vector |b|, that already contain the inequalities from above. We wish to enforce that DER3 and DER4 are no more than one AREA (RUA) from each other, and similarly for DER5 and DER6. First the symmetric distance matrix for the RUAs is built using

 ISBN: 978-1-61208-107-6

physical locations and Manhattan (i.e., the “taxicab” metric).
D = zeros(numAREAS); // generates a 7 x 7 zero matrix
Setting up the top right half of the matrix
D(1,2:end) = [1 2 3 2 3 4];
D(2,3:end) = [1 2 1 2 3];
D(3,4:end) = [1 2 1 2];
D(4,5:end) = [3 2 1];
D(5,6:end) = [1 2];
D(6,end) = 1;
The lower left half is the same as the upper right D = triu(D)' + D; We find the RUA’s that are more than one distance unit away.

```
>> D

D =

     0     1     2     3     2     3     4
     1     0     1     2     1     2     3
     2     1     0     1     2     1     2
     3     2     1     0     3     2     1
     2     1     2     3     0     1     2
     3     2     1     2     1     0     1
     4     3     2     1     2     1     0

>> D = triu(D)' + D;
>> D

D =

     0     1     2     3     2     3     4
     2     0     1     2     1     2     3
     4     2     0     1     2     1     2
     6     4     2     0     3     2     1
     4     2     4     6     0     1     2
     6     4     2     4     2     0     1
     8     6     4     2     4     2     0
```

[AREAA,AREAB] = find(D > 1);
numPairs = length(AREAA);
This finds |numPairs| pairs of AREAS. For example, if DER3 occupies one AREA in the pair, then DER4 cannot occupy the other AREA in the pair, else it would be more than one unit away in terms of AREA. The same condition holds for DER5 and DER6. This gives |2*numPairs| additional inequality constraints which we add to |A| and |b|. By adding rows to A, we accommodate these constraints as follows:
numrows = 2*numPairs + numAREAS;
A((numAREAS+1):numrows, 1:numDim) = zeros(2*numPairs,numDim);
For each pair of AREAS in numPairs, for the |x(i)| that corresponds to DER 3 in |AREAA| and for the |x(j)| that corresponds to DER4 in |AREAB|, x(i) + x(j) <= 1 i.e., either DER3 or DER4 can occupy one of these AREAS, but not both.

C. Branch and Bound (BB) Strategy

The branch and bound algorithm is a well-known optimal solution method. Branch and bound (BB) algorithms are methods for solving non-convex global optimization problems [6-8]. They are exact (non-heuristic), in the sense that they calculate a provable upper and lower bounds on the globally optimal objective value and they terminate when all suboptimal feasible solutions have been eliminated. Branch and bound (BB) algorithms can be computationally slow. In the worst case they require effort that grows exponentially with problem size. We achieved fast convergence in our problems. We do note that due to total unimodularity of the basic A matrix, that a network-based customized linear programming solver could be used to provide the lower bounds very quickly in large problems. The BB algorithm is a well known algorithm in the research community [6-9]. An example run of the Branch and Bound algorithm is shown in Fig.5 followed by a snippet of MATLAB code showing the iterative output for each node displayed in the branch and bound algorithm. We let the BINTPROG choose the start point.

x0 = [];
f = -c;
options = optimset('Display','iter','NodeDisplayInterval',1);
[x,fval,exitflag,output] =
bintprog(f,A,b,Aeq,beq,x0,options);
fval
exitflag
output

To reduce the number of nodes explored, the time, or number of iterations taken, there are alternative options available. BINTPROG use the options to adjust the algorithm with differing node and branching variable strategies [2].

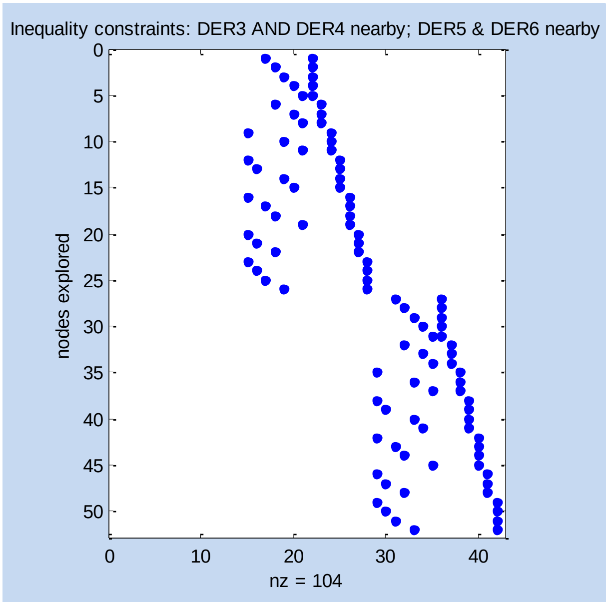


Figure 5. BB search algorithm with Inequality constraints

 ISBN: 978-1-61208-107-6

For example, the default branching strategy is |'maxinfeas'|, which chooses the variable with the maximum integer infeasibility for the next branch, that is, the variable whose value is closest to 0.5. Running the problem again with the branching strategy set to |'mininfeas'|, the variable the minimum integer infeasibility is chosen (that is, the variable whose value is closest to 0 or 1 but not equal to either).

For structuring the tree, depth-first and best-node search strategy are available. For example, in 'df', at each node in the search tree, if there is a child node one level down in the tree that has not already been explored, the algorithm chooses one such child to search. Otherwise, the algorithm moves to the node one level up in the tree and chooses a child node one level down from that node. In best-node (bn) strategy, the node with lowest bound on the objective function is the default. In our limited computational experience, convincing and acceptable results were quickly reached. For future work, we would plan to increase the scale of our test problems and investigate improved BB schemes.

V. RESULTS

The simulation is carried out in a MATLAB platform. The results show that the optimal value is reached after 163 iterations with 54 nodes participation in 1.22 seconds (case1) using the capacity based Iterative Binary Integer Linear Programming (C-IBILP) based branch and bound method which maximizes the satisfaction of the DER preferences weighted by its capacities.

The final output shown in Figure 6 presents the DER allocation with the RUA 1 or area 1 treated as empty for optimal assignment.

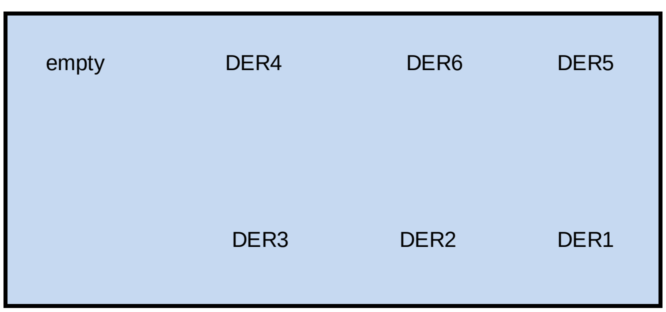


Figure 6. An optimal DER assignment solution for case 1

Case 2: If we change the preferences of DER's according to the matrix shown below, then the optimal solution is reached with 13 iterations, 1 node in 0.047 seconds with default node and branch strategies.

```
>> prefmatrix

prefmatrix =

     0     0     0     1     3    70
     0     0     0     3    40    10
     0     0     0     3     1    10
     0     0    70     3     2    10
    10    90    30    10    10    20
    40    40    40    60    40    20
    50    40    30    40    40    20
```

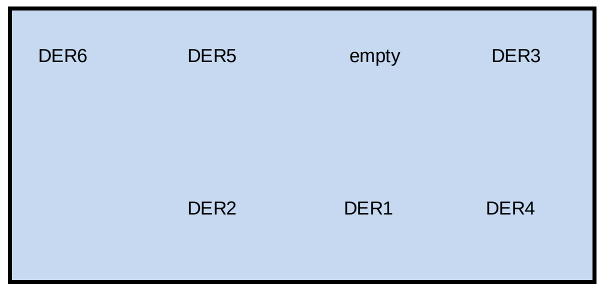


Figure 7. An optimal DER assignment solution to case 2

VI. CONCLUSION

The paper presents a resource assignment problem for smart grid application. The capacity based Iterative Binary Integer Linear Programming (C-IBILP) model is designed to specify an optimal allocation of distributed energy resources (DER's) during power outage periods to satisfy shortages. Computational results show that our C-IBILP algorithm exhibits very good performance for problem instances tested. A branch and bound algorithm for the smartgrid problem was described. It combines the extension results previously presented in the literature with new elements, such as a new lower bound that works by exploiting some properties connected with the ad- hoc branching rule we have developed. Computational results establish that the algorithm is very competitive. It greatly improves the results obtained by methods that have recently appeared in the literature. The limitation of our approach is that the method does not scale well for larger DERs. Our current efforts involve extending this assignment model to a more scalable assignment formulation for which larger numbers of DER's can to serve each RUA.

VII. REFERENCES

[1] M. Amin, "Balancing market priorities with security issues: interconnected system operations and control under the restructured electricity enterprise," IEEE Power and Energy Magazine, vol. 2, pp. 30-38, July/August 2004.

 ISBN: 978-1-61208-107-6

[2] Mathworks Inc, Binary Integer Programming (BINTPROG), http://www.mathworks.com, accessed 08.31.2010.
[3] S. Arora, B. Bollobas, L. Lorasz, and I. Tourlakis, "Proving integrality gaps without knowing the linear program," Theory of Computing, vol. 2, pp. 19-51, February, 2006.
[4] M. Alekhnovich, S. Arora, and I. Tourlakis, "Towards strong nonapproximability results in the Lovasz-Schrijver hierarchy." Proc. of the 37th Annual ACM Symp. on Theory of Computing, ACM, 2005, pp.294–303, doi:10.1145/1060590.1060634.
[5] C. Barnhart, E. Johnson and G. Nemhauser,1998. "Branch-and-Price: Column generation for solving huge integer programs," Operations Research vol. 46, 1998, pp.316–329.
[6] B. Balakrishnan, and S. Balemi. "Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems," Int. J. of Robust and Nonlinear Control, vol. 1, 1991, pp.295–317, October–December 1991.
[7] E. Lawler and D. Wood. "Branch-and-bound methods: A survey," Operations Research, vol 14, 1966, pp.699–719.
[8] R. Moore. Global optimization to prescribed accuracy. Computers and Mathematics with Applications, vol. 39, 1991, pp.25–39.
[9] L. Wolsey, Integer Programming, John Wiley & Sons, 1998.
[10] G. Nemhauser and L. Wolsey, Integer and Combinatorial Optimization, John Wiley & Sons, 1988.
[11] F. Hillier, Frederick. and G. Lieberman., Introduction to Operations Research, McGraw-Hill, 2001.
[12] http://www.oe.energy.gov/smartgrid.htm, accessed 08.31.2010.
[13] P. Kouvelis and P. Yu. Robust Discrete Optimization and its Applications. Kluwer Academic Publishers, 1997.
[14] G. Kozina and V. Perepelista. "Interval spanning trees problem: solvability and computational complexity," Interval Computations, vol. 1, 1994, pp.42-50.
[15] J. Kruskal. "On the shortest spanning subtree of a graph and the travelling salesman problem," Proc. of the American Mathematical Society, 1956, pp.48-50.
[16] R. Montemanni and L. Gambardella. "A branch and bound algorithm for the robust spanning tree problem with interval data," European Journal of Operational Research, vol. 161, March, 2005, pp.771-779.
[17] S. Khot and N. Vishnoi, "The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics," Proc. of the 46th Symposium on Foundations of Computer Science, 2005, pp. 53-63, doi:10.1109/SFCS.2005.74.

 ISBN: 978-1-61208-107-6

An Integrated VLE to Enhance Social Interaction for User-centred Active Learning

Li Jin
School of Electronics and Computer Science
University of Westminster, UK
L.Jin02@westminster.ac.uk

Zhigang Wen
Imagination Technologies Ltd.
Kings Langley, Watford
Hertfordshire, UK

***Abstract*—With the advent of social networking on the Internet, the great impacts of social interaction have rapidly migrated into the online world in order to enable multiple users to share opinions, insights, expertise, experiences, and interests with each other and further reduce the barriers to collaboration, skill-development, and discovery. Social networking facilitates social interaction and share of user-generated content in a collaborative environment. It has transferred the Internet into a dynamic platform for the development of innovative e-commerce and e-learning applications. Social networking has already begun to foster an intuitive and immersive virtual environment by converging Web 3D technology for enhancing user motivation and engagement. This paper presents an integrated Virtual Learning Environment (VLE) which is designed to incorporate social networking services into conventional VLEs in order to support both informal and formal learning practices required in Higher Education. A Social Virtual World (SVW) is adopted as a visual interface to access the integrated VLE since it enhance social interaction for user-centred active learning by supporting visual communication, mutual awareness and accountability. This paper further discusses and experiments some actual educational practices prototyped in the proposed integrated VLE to evaluate the valued learning experience achievement.**

Keywords- Social interaction; e-Learning; Virtual Learning Environments

I. INTRODUCTION

Humans are fundamentally social beings bounded to interact with each other by establishing empathy relations, to create groups, and to collaborative with others. Social Interaction usually follows a set of predefined behavioural patterns or models shaped by our natural abilities and rational skills to improve productivity, especially when a group of people works in a given homogeneous cultural environment [1]. Therefore, the social information obtained through interaction plays an important role in decision making and collaborative working. For instance, people would more likely purchase the best selling products (e.g. books, games, and toys) in real life because the best selling product implies that many people who have used it were satisfied with it and even were inspired by it. With the advent of social networking on the Internet, the great impacts of social interaction have rapidly migrated into the online world in order to enable multiple users to share opinions, insights, expertise, experiences, and interests with each other and further reduce the barriers to collaboration, skill-development, and discovery.

Social networking facilitates social interaction and share of user-generated content in a collaborative environment. It has evolved and transferred the Internet into a platform for the development of innovative e-commerce and e-learning applications [2, 3, 4, 5]. Social networking has already begun to foster an intuitive and immersive virtual environment by converging Web 3D technology. As one of emerging technologies, Social Virtual worlds (SVWs) has expanded and challenged ideas of the next generation of virtual learning environments. It is important for educators to analyze and understand what the characteristics of the 21st century learners are and how learning is changing as a result of these learners' participating in these environments in comparison to the conventional static and text-oriented Virtual Learning Environments (VLEs).

In this paper, Section II firstly analyzes multimodal interaction and requirements of social interaction enhancement through social networking. It then discusses the impact of SVWs on the Internet and its applications for e-learning. Section III outlines the limitations of SVWs when being used in formal educational learning and then presents an integrated VLE which is designed to incorporate social networking services into conventional VLEs in order to support both informal and formal learning practices. A SVW is adopted as a 3D visual interface to access the integrated VLE since it facilitates social interaction for user-centred active learning by supporting visual communication, mutual awareness and accountability. Section IV discusses and experiments some actual educational practices prototyped in SVWs to evaluate the learning experience achieved in the proposed integrated VLE. Finally, a brief conclusion is drawn in section V.

II. ANALYSIS AND REQUIREMENTS

Social networking on the Internet provides a dual-way mechanism for users, which allows them not only to read and download content in a top-down approach as before but also to write and upload online content in a bottom-up approach.

 ISBN: 978-1-61208-107-6

By using user friendly interfaces along with rich media, it encourages users as producers to enrich information, including video, audio and even 3D data. This stands in contrast to traditional Internet use, which limits users to browsing content that only the site owner can modify in a top-down approach. The advent of social networking is transferring the Internet into a dynamic platform for user active participation rather than a primary conventional information repository. The impact of the emerging technology has resulted in facilitating social interaction and share of user-generated content in a collaborative online environment. This section analyzes multimodal interaction on social networking and identifies the requirement of social interaction enhancement. It then discusses the impact of SVWs on the Internet and its applications for e-learning.

A. Social Interaction on Social Networking

As a novel medium, social networking facilitates multimodal interaction among users to improve information exploration and enrichment in collaborative way. Different social networking services focus on different aspects of human interaction. For example, MySpace (www.myspace.com) as a media-based social network allows self-publishing within users' network of friends and colleagues. Facebook (www.facebook.com) allows users to create personal profiles and build up social relationship with other users by uploading various media such as photos and videos. Wikis provide multiple users a web-based collaborative interface to edit content and add links. Blogs support regular and frequent content editing via quickly thoughts and images posting and interaction with the public. It provides a quick, responsive, and user-centred mechanism for effective collaboration. YouTube (www.youtube.com) as an online video-sharing network allows users to upload and share videos clips and give out comments on shared content.

One of the clear evolutionary trends of social networking is that these sites keep changing because there is a constant drive for extension and development as social grouping change and reformation as a result of social interaction among large numbers of users through these services. For example, while browsing a discussion forum, users may select a particular review or comment with a high rating from other users. Such decision is actually made based on observation of the activity of other users in the information space. To support coherent interaction in online environments, Erickson and Kellogg in IBM research centre developed a social translucence approach [6] which aimed at revealing and visualizing the presence and activity of users in digital environments in order to promote shared awareness and to support social processes. It emphasized the importance of perceptually-based social cues visible to their users and pointed out three key features of such systems for social interaction enhancement [7]:

- Visibility/Co-presence: A system should allow users to observe each others' presence and actions and further offer a sense of co-presence - "being there together" which is bound to be closely related. In face-to-face settings, social information is used to guide interaction among users in the system.
- Awareness: A system enhances mutual awareness among the users through their co-presence. However, this awareness does not mean that user activities becomes total transparent. A system must protect user privacy and only reveal information that is necessary to support social awareness.
- Accountability: Shared awareness leads to user accountability. Since users know that they can see each others' actions, they must be responsible for their actions. This facilities user to follow social rules and conventions.

However, it is observed that most social networking services on the Internet are still heavily based on text, image and video such as blogs, wikis, Facebook, and YouTube. There is a lack of efficient ways to support rich interaction that mainly enhanced by perceptually-based social cues visible among users such as gesture, posture, and emotional expression in real time. Recently, in conjunction with Web 3D technology, social networking has already begun to foster an intuitive and immersive 3D Social Virtual World (SVW) to address these deficiencies.

B. Social Virtual World for E-learning

Social Virtual Worlds (SVWs) are intended to build up lively-based interactive virtual communities which represent part of reality and also leave some space for fantasy to be filled in. Users interact with each other through their emotional avatars in a 3D virtual world. One of the most successful SVWs is Second Life (www.secondlife.com), an online social space in which users can explore, meet others, socialize, and participate in individual and group activities for educational or business purposes. Since its introduction to public in 2003, the 3D virtual community has grown explosively and today is inhabited by millions of users from all around the world. Some well-known companies start to embed SVWs into e-commence and e-performance services [8, 9]. For example, the British Broadcasting Corporation (BBC) staged its annual 'One Big Weekend' rock concert in Second Life. Online audiences were able to see avatars of their favorite musicians, as well as watch and listen to live streams of the bands on stage in Scotland. It is agreed SVWs added a social level of interactivity for those who were unable to attend physically. Meanwhile, thanks to the popularity and success of massively multiplayer online role-playing game (MMORPG), SVWs are rapidly growing in popularity due to game-play features. These massive 3D virtual environments have been widely accepted by wide range of people and become more attractive for young generation. Millions of users spend hours in SVWs at a time socializing, competing, and most of all, learning, learning how to builds digital creative content (e.g. architecture and crafts), learning how to work as a team, learning how to make decisions, and learning how to solve problems. They enjoy engaging within learning activities and are highly motivated and stimulated in the environments.

 ISBN: 978-1-61208-107-6

While offering a source of entertainment for users, SVWs make huge contribution to user-centred active learning. They have begun to be used in e-learning by many institutions, such as colleges, universities, libraries, and government entities. Institutions explore SVWs for a wide range of educational activities including learning, teaching and research in order to enhance personal development skills within a collaborative and sharing virtual environment [10, 11, 12]. Researchers and educators favour the innovative learning environment because it is more personal and social than traditional e-learning. In recent, Second Life has become one of novel virtual learning platforms for major colleges and universities, including the University of Florida, Harvard, Open University, and Ohio University.

III. DESIGN OF AN INTEGRATED VLE

Wegerif [13] emphasizes that many evaluations of asynchronous learning networks understandably focus upon the educational dimension, either learning outcomes or the educational quality of interaction, overlooking the social dimension. After identifying the requirement of social interaction enhancement and investigating SVWs for e-learning, we believe that SVWs should be adopted as a means to extend and enhance the existing VLE for user-centred active learning. Users are encouraged to become dependent on each other's knowledge and experience to the point of leading and teaching others. However, SVWs along with other social networking services have primarily made contribution to informal learning which is mainly relies on self-motivation and self-management. There are still some challenges of adopting social networking for formal learning in High Education (HE) systems which requires explicit curriculum design, indicative content development, learning material management, and learning outcome achievement [4].

This section presents the design of an integrated social learning approach for user-centered active learning in order to support both informal and formal learning practices. The approach is to propose an integrated VLE system on which enables incorporating SVWs as a new component into traditional distance learning. Figure 1 illustrates this integrated VLE which is designed to use a SVW as a visualization integration interface to present educational activities while coupling with conventional Learning Management System (LMS) such as Blackboard and student supporting system.

The integrated system is intended to carry out learning in SVWs through both formal and informal practices including seminars, workshops, group discussions, and social events which are organized and led by guided members or through peer-to-peer synchronous voice chatting, or instant messaging (IM). A SVW (e.g. Second Life) is adopted as a visual interface to access the integrated VLE for user-centred active learning since it facilitates social interaction to improve learners' motivation and engagement in learning activities by supporting visibility, mutual awareness and accountability. Such a system will make it easier for users to carry on coherent discussions; to observe and imitate others' actions; to engage in peer pressure; to create, observe, and

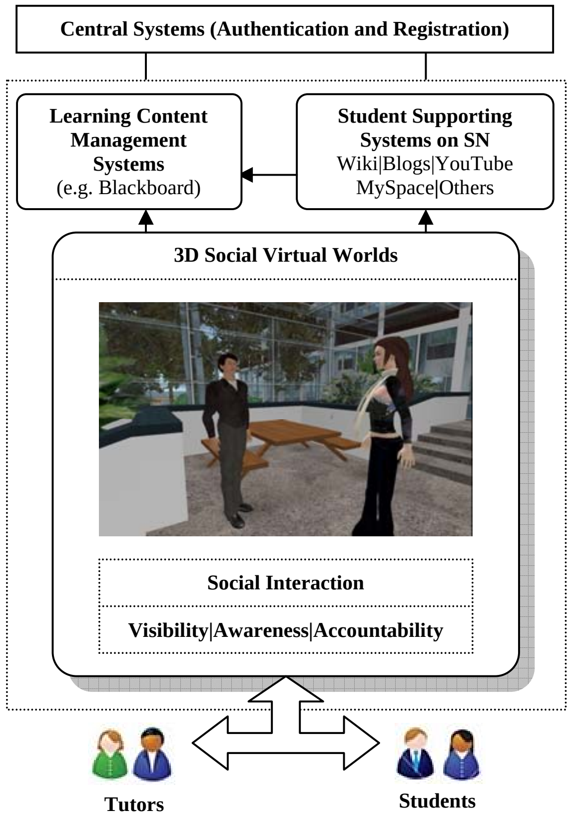


Figure 1. The integrated VLE enhanced by social interaction

conform to social conventions; and to engage in other forms of collective interaction as suggested in the social translucence approach by Erickson and Kellogg [6].

The conventional LMS such as Blackboard is responsible for hosting courses on the Internet as a supplement to traditional classroom courses. It is used for course management in this VLE and allows lecturers and tutors to upload and organize course material including course content, reading lists, assessment, and announcement similar to those used in classroom. Students can browse and download these course materials via the LMS. The integrated system relies on a customizable open architecture, authentication protocols, and a scalable design that allows the integration of other social networking services such as YouTube, blogs, and wikis in student supporting systems. These social networking services are interlinked within a SVW (e.g. Second Life) to enhance the efficiency of communication between tutors and students and further achieve real-time social interaction among students and tutors. Learning in the integrated VLE is supported by exchange of knowledge, expertise, and information through coupling with these social networking services such as writing blogs, co-editing wikis, uploading videos on YouTube, or joining MySpace and Facebook.

The integrated system allows students and tutors to download or upload learning content, and conduct course-related discussion through a 3D space that students would

 ISBN: 978-1-61208-107-6

like to take part in, where they not only work on course-related tasks, but also relax, socialize, and talk with others. It enables connecting students and tutors on both an academic level and a social level for supporting both formal and informal learning.

IV. EXPERIMENTS AND EVALUATION

In order to evaluate the proposed system, this section discusses and experiments some actual learning practices prototyped in the integrated VLE. In our research, a virtual campus build up in Second Life as a visual interface of the integrated VLE allows users to access a social interactive learning environment where learning practices occur including workshops, seminars, invited speeches, group discussions, and tutorial demonstrations. The virtual campus represents learning scenarios in 3D intuitive way and uses hypermedia to improve delivery of learning materials beyond text. It has capability of interlinking with LMS such as Blackboard in conventional e-learning systems and integrating with other social networking tools such as YouTube, blogs, and wikis. The methods followed to deliver lectures in a SVW are very similar to those followed in the real world like PowerPoint presentations, video tutorials, and guest lectures. But at the same time there are facilities that would be possible only in a SVW. Since educational activities are diverse, the following learning practices are chose to be prototyped in SVWs for experimental study.

1) Role Playing Simulations: Role-playing simulations have been popular teaching and learning methods which encourage students to enact different roles through acting rehearsals. Role-playing activities help students familiarize the situations they might encounter in their work sectors and therefore have been used extensively for training. SVWs introduce a significant shift in student-centre active learning because students are able to take on new roles in such a game-like virtual environment or expand their identities as a supplement to a real world identity. Also they have great impacts on the change of the roles of students and instructors. Students in SVWs present social information about their enacted roles and further develop their identity and knowledge both outside and inside the world by role playing. One such example would be the production of a court simulation in the virtual campus where the avatars participate in the role play involving lawyers, interest groups, and justices. This type of role-playing activity requires that students apply their substantive knowledge of constitutional law and the jurisprudence of their justice in deciding important constitutional issues. Although the simulation places extra demands on both instructors and students, the effort is worthwhile because this kind of training becomes a challenging social interactive learning practice.

2) Social Events: Many academic social events are organised regularly in universities in order to enhance the students' active learning experience. By taking advantage of SVWs, we launched a virtual exhbition in our virtual campus as shown in Figure 2. The feature of 'the learner as content producer' in a SVW allows students to put their visual works (e.g. poster, animation, video) as user-generated content in the virtual exhbition centre. Students can change and replace their exhibits whenever they want. Compared to the on-site gallery show, the virtual exhibition in a SVW has great competitive advantages including low cost, removing the barriers such as limited time and space of exhibition, and being able to reach a much wider audience through the Internet. Furthermore, the learner as content producer in this SVW has greatly spurred students' motivation, enhancing their creativity and productivity. As a result, students are increasingly stimulated into active participation in social events in the intergrated VLE while achieving a student-centred active learning experience.

Figure 2. The virtual exhibition event

3) E-mentoring/e-tutoring: Previous research pointed out that successful mentoring demands frequent and regular interaction but all sorts of barriers such as time, work responsibilities, geographical distance and lack of trust - often reduce if not halt interaction [14]. In the case of distance learning when students are geographically dispersed, it is usually impossible for mentors and mentees to meet face-to-face regularly. A SVW as an visual interface in the integrated VLE is used to teleport students to the virtual campus. An e-mentoring system was developed in order to allow students to discuss their work with their online tutors or mentors through their 3D avatars by real time communication in the form of voice chatting, IM, and even non-verbal expressive gesture and posture. For example, as shown in Figure 3, one of our MSc students joined in an online tutorial in Second Lift while he was actually in Spain and met up his tutor based in the UK. With the utilisation of a SVW, an e-tutoring/e-mentoring system offers cost-effective solutions which impinge less upon the participants' time and effort, so that more frequent social interaction is easier to achieve and manage. It has several

 ISBN: 978-1-61208-107-6

advantages, including open correspondence, access to more geographically isolated regions, and efficiency of communication. As a result, students feel that the e-tutoring/e-mentoring systems in the SVW are more attractive and engaging, enabling more thoughtful and deliberate discussions with trust.

Figure 3. E-mentoring/E-tutoring

According to Gilroy [17], the formula of Valued Learning Experience = F (Pedagogy, Trust, Content, Community). This emphasized the key factors for achieving valued learning experience include a functional pedagogy for instruction, the level of trust, indicative content to be learned, and a networked community for collaborative learning. Based on the above experimental study, it is observed that the integrated VLE coupled with a SVW as visual interface has great potential to support user-centred active learning in terms of the creation, distribution, and access of learning resources, collaboration and interaction, time and location independency, roles changing (e.g. student and tutor), and learning outcomes achievement. There are many other learning practices such as workshops, seminars, and conferences, and even alumni reunions that can be prototyped and take place in the virtual campus. Based on the overall positive student feedback for these educational applications, the social advantages and student-centred learning experience offered by the integrated learning environment are seen to be strongly valued to students. With the utilization of SVW as a social portal, the integrated VLE not only allow users with specific learning requirements to be able to access and share of learning materials from dispersed locations through a visual online interface but also offer the following innovative characteristics for being a social learning environment:

- Massively multi-user participation: to enable a large number of learners joining together in shared 3D virtual environments and be able to attract new learners.
- Focus of socialisation and collaboration: to focus on interpersonal relations rather than moving around in the environment and further encourage the formation of in-world social groups in order to coordinate and work together for a common learning objectives.
- Multimodal social interaction: to offer rich modes of real-time interaction based on perceptually-based social cues such as gesture, posture, and emotional expression to support visual communication, mutual awareness and accountability.
- Focus of self-motivation and active engagement: to facilitate self-motivated participation and attract a number of learners to actively take part in a focus group because of common learning interest.
- Share of learner-generated content: to support learners to contribute customized content and encourage sharing of learning materials generated by learners themselves as creative effort.
- Low cost participation: no fee for registration and low cost participation in SVWs.

V. CONCLUSIONS

By introducing social interaction into e-learning that conventional VLE cannot offer, it is seen that SVWs can be used to enhance various types of learning practices for user-centred active learning. This paper has presented an integrated VLE which was designed to incorporate social networking services into conventional VLEs in order to support both informal and formal learning practices required in Higher Education. A SVW is adopted as a visual interface to access the integrated VLE since it facilitates social interaction to improve learners' motivation and engagement in learning activities by supporting co-presence, mutual awareness, and accountability. The integrated VLE has considerable advantages and strength to benefit users, including improving student engagement and motivation, offering student-centred active learning experience, and introducing socio-technical innovations. Preliminary experiments prototyped in the integrated VLE, including social events, e-tutoring/mentoring, and role-playing have produced favourable responses from academics and students. In summary, SVWs are transforming the nature of learning as social practice in a collaborative environment. Through enhancing social interaction, the integrated VLE will be able to facilitate self-motivation, active engagement, and creative thinking in user-centred active learning. More experimental educational activities will be validated in future work and these will be further evaluated through student feedback and progression analysis. It is intended to expand the social interactive VLE by integrating more academic resources and services departments in HE such as career service, library service and training centres. However, social networking also raises challenging research issues about privacy, identity and Intellectual Property (IP) [11] and even after-effects issues including how online behaviours affect users' behaviour offline in real life. Furthermore, higher level interaction (e.g. cognitive, motivational) has not yet been fully achieved in SVWs and researchers and practitioners are now attempting to address these complex natural multimodal interaction issues [16].

ACKNOWLEDGMENT

This project is supported by Education Initiative Centre at the University of Westminster. Thanks to the contribution of our MSc Computer Animation Students to the development of virtual campus in Second Life.

REFERENCES

[1] Quemada, J., “Collaboration, Social Interaction and Social Protocols”, Proccedings of WTICE’08, Italy, 23-25 June.

[2] Churchill, E.F, and Halverson, C.A., “Social Networks and Social Networking”, IEEE Internet Computing, vol. 9, issue 5, Sep/Oct 2005, pp.14-19.

[3] Wever, B.D., Mechant, P., Veevaete, P., and Hauttekeete, L., “E-Learning 2.0: social software for educational use”, Proceedings of 9th IEEE International Symposium on Multimedia, Taiwan, 10-12 Dec, 2007, pp.511-516.

[4] Marsick, V.J. and Watkins, K.E., “Informal and Incidental Learning”, New Directions for Adult and Continuing Education, issue 89, Wiley Periodicals, 2001, pp.25-34.

[5] Lockyer, L. and Patterson, J., “Integrating Social Networking Technologies in Education: A Case Study of a formal learning environment”, Proceedings of 8th IEEE International Conference on Advanced Learning Technologie, Santander, Spain, 1-5 July 2008, pp.529-533.

[6] Erickson, T., and Kellogg, W.A., “Social Translucence: Using Minimalist Visuallisations of Social Activity to Support Collective Interaction”, Designing Information Spaces: The Social Navigation Approach, 2003, pp. 75-105.

[7] Vassileva, J., and Vassileva, I., “Social Interaction History, A Framework for Supporting Exploration of Social Information Spacess”, Proceedings of 2009 International Conference on Computer Science and Engineering, Vancouver, Canada, 29-31 Augues, pp.538-545.

[8] Dell 2009, “Dell Island in SecondLife”, http://www.dell.com/html/global/topics/sl/index.html, last accessed Feb 2009.

[9] Hetherington, R., Bonar-Law, J., Fleet, T., and Parkinson, L., “Learning in a Multi-User Virtual Environment”, Proceedings of IEEE 2008 International Conference Visualization, Columbus Ohio, USA, 19-24 Oct. 2008, pp.99-105.

[10] Cunha, M., Raposo, A., and Fuks, H., “Educational Technology for Collaborative Virtual Environments”, Proceedings of 12th International Conference on Computer Supported Cooperative Work in Design, Xian, China, 16-18 April, 2008, pp. 716-720.

[11] Hendaoui, A., Limayem, M., and Tompson, C.W., “3D Social Virtual Worlds, Research Issues and Challenges”, IEEE Internet Computing, vol.12, issue 1, Jan/Feb 2008, pp.88-92.

[12] Araujo, R. M., Rezende, E.A., Andrade, T.S., Chaves, V.M., Lopes, M.G., and Diirr, B., “People in Network, Collaboration for Action: New Supporting Requirements”, Proceedings of 11th International conference on Computer Supported Cooperative Work in Design, Melbourne, Australia, 26-28 April, 2007, pp.939-944.

[13] Wegerif, R., “The Social Dimension of Asynchronous Learning Networks”, Journal of Asynchronous Learning Networks, Vol.2, issue 1, pp 34-42.

[14] Bierema, L, and Meriam, S., “ E-mentoring: Using Computer Mediated Communication to Enhance the Mentoring Process, Innovative Higher Education, Vol.26, No 3, 2002, pp211-227.

[15] Weaver, A.C. and Morrison, B.B., “Social Networking”, IEEE Computer, vol. 41, issue 2, Feb. 2008, pp.97-100.

[16] Breslin, J. and Decker, S., “The Future of Networks on the Internet, the Need for Semantics”, IEEE Internet Computing, vol. 11, issue 6, Nov/Dec 2007, pp 86-90.

[17] Gilroy, K, “Collaborative E-learning: the Right Approach”, ArsDigita Systems Journal, www.arsdigita.com/asj/elearning/, last accessed 18 March 2010.