Tran Khanh Dang · Roland Wagner
Josef Küng · Nam Thoai
Makoto Takizawa · Erich Neuhold (Eds.)

LNCS 10018

Future Data and Security Engineering

Third International Conference, FDSE 2016
Can Tho City, Vietnam, November 23–25, 2016
Proceedings

FDSE 2016

Springer

Lecture Notes in Computer Science 10018

More information about this series at http://www.springer.com/series/7409

Tran Khanh Dang · Roland Wagner
Josef Küng · Nam Thoai
Makoto Takizawa · Erich Neuhold (Eds.)

Future Data and Security Engineering

Third International Conference, FDSE 2016
Can Tho City, Vietnam, November 23–25, 2016
Proceedings

Springer

Editors

Tran Khanh Dang
University of Technology
Ho Chi Minh City
Vietnam

Nam Thoai
University of Technology
Ho Chi Minh City
Vietnam

Roland Wagner
Johannes Kepler University
Linz
Austria

Makoto Takizawa
Hosei University
Tokyo
Japan

Josef Küng
Johannes Kepler University
Linz
Austria

Erich Neuhold
University of Vienna
Wien
Austria

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-48056-5 ISBN 978-3-319-48057-2 (eBook)
DOI 10.1007/978-3-319-48057-2

Library of Congress Control Number: 2016954943

LNCS Sublibrary: SL3 – Information Systems and Applications, incl. Internet/Web, and HCI

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

In this volume we present the accepted contributions for the Third International Conference on Future Data and Security Engineering (FDSE 2016). The conference took place during November 23–25, 2016, in Can Tho City, Vietnam, at Can Tho University of Technology, which is the first technical-oriented university in the Mekong Delta area. The proceedings of FDSE are published in the LNCS series by Springer. Besides DBLP and other major indexing systems, FDSE proceedings have also been indexed by Scopus and listed in the Conference Proceeding Citation Index (CPCI) of Thomson Reuters.

The annual FDSE conference is a premier forum designed for researchers, scientists, and practitioners interested in state-of-the-art and state-of-the-practice activities in data, information, knowledge, and security engineering to explore cutting-edge ideas, present and exchange their research results and advanced data-intensive applications, as well as to discuss emerging issues on data, information, knowledge, and security engineering. At the annual FDSE, the researchers and practitioners are not only able to share research solutions to problems of today's data and security engineering themes, but also able to identify new issues and directions for future related research and development work.

The call for papers resulted in the submission of 115 papers. A rigorous and peer-review process was applied to all of them. This resulted in 27 full (including keynote speeches) and two short accepted papers (acceptance rate: 25.2 %), which were presented at the conference. Every paper was reviewed by at least three members of the international Program Committee, who were carefully chosen based on their knowledge and competence. This careful process resulted in the high quality of the contributions published within this volume. The accepted papers were grouped into the following sessions:

- Big data analytics and cloud data management
- Internet of Things and applications
- Security and privacy engineering
- Data protection and data hiding
- Advances in authentication and data access control
- Access control in NoSQL and big data
- Context-based data analysis and applications
- Emerging data management systems and applications

In addition to the papers selected by the Program Committee, four internationally recognized scholars delivered keynote speeches: "The Present and Future of Large-Scale Systems Modeling and Engineering," presented by Prof. Dirk Draheim from Tallinn University of Technology, Estonia; "Internet of Things and Reasons Why It Is Becoming a Reality," presented by Prof. Cong-Duc Pham from the University of Pau, LIUPPA Laboratory, France; "Knowledge Processing and Security Aspects for the Agricultural Domain," presented by Prof. Josef Küng from Johannes Kepler University Linz, Austria;

and “Study on the Difference of Ecological Cognition Between the Real Environment and the Virtual Environment and Its Compensation,” presented by Prof. Kazuhiko Hamamoto from Tokai University, Japan. Besides, we also organized tutorials from both academic institutes as well as industrial sectors.

The success of FDSE 2016 was the result of the efforts of many people, to whom we would like to express our gratitude. First, we would like to thank all authors who submitted papers to FDSE 2016, especially the invited speakers for the keynotes and tutorials. We would also like to thank the members of the committees and external reviewers for their timely reviewing and lively participation in the subsequent discussion in order to select such high-quality papers published in this volume. Last but not least, we thank Can Tho University of Technology, and the Faculty of Computer Science and Engineering, HCMC University of Technology, for hosting and organizing FDSE 2016.

November 2016

Tran Khanh Dang
Roland Wagner
Josef Küng
Nam Thoai
Makoto Takizawa
Erich Neuhold

Organization

General Chairs

Roland Wagner	Johannes Kepler University Linz, Austria
Abdelkader Hameurlain	Paul Sabatier University, Toulouse, France

Steering Committee

Elisa Bertino	Purdue University, USA
Kazuhiko Hamamoto	Tokai University, Japan
Koichiro Ishibashi	The University of Electro-Communications, Japan
M-Tahar Kechadi	University College Dublin, Ireland
Dieter Kranzlmüller	Ludwig Maximilians University, Germany
Josef Küng	Johannes Kepler University Linz, Austria
Clavel Manuel	The Madrid Institute for Advanced Studies in Software Development Technologies, Spain
Fabio Massacci	University of Trento, Italy
Atsuko Miyaji	Osaka University and Japan Advanced Institute of Science and Technology, Japan
Benjamin Nguyen	Institut National des Sciences Appliqués Centre Val de Loire, France
Beng Chin Ooi	National University of Singapore, Singapore
Silvio Ranise	Fondazione Bruno Kessler, Italy
Nam Thoai	HCMC University of Technology, Vietnam
A Min Tjoa	Technical University of Vienna, Austria
Shigeki Yamada	National Institute of Informatics, Japan

Program Committee Chairs

Tran Khanh Dang	HCMC University of Technology, Vietnam
Makoto Takizawa	Hosei University, Japan
Erich Neuhold	University of Vienna, Austria

Publicity Chairs

Phan Trong Nhan	Johannes Kepler University Linz, Austria
Nguyen Duc Dung	HCMC University of Technology, Vietnam
Quan Thanh Tho	HCMC University of Technology, Vietnam
Hoang Tam Vo	IBM Research, Australia

Local Organizing Committee

Duong Thai Cong	Can Tho University of Technology, Vietnam (Co-chair)
Tran Khanh Dang	HCMC University of Technology, Vietnam (Chair)
Josef Küng	Johannes Kepler University Linz, Austria
Lam Son Le	HCMC University of Technology, Vietnam
Hong Thanh Luan	Can Tho University of Technology, Vietnam
Thanh Binh Nguyen	HCMC University of Technology and Can Tho University of Technology, Vietnam
Nguyen Thi Xuan Thu	Can Tho University of Technology, Vietnam
Truong Minh Nhat Quang	Can Tho University of Technology, Vietnam
Tran Minh Quang	HCMC University of Technology, Vietnam
Nguyen Thanh Tung	HCMC University of Technology, Vietnam
Truong Quynh Chi	HCMC University of Technology, Vietnam
Tran Thi Que Nguyet	HCMC University of Technology, Vietnam
Le Thi Kim Tuyen	HCMC University of Technology, Vietnam

Finance Chairs

La Hue Anh	HCMC University of Technology, Vietnam
Ngo Quoc Huu	Can Tho University of Technology, Vietnam

Program Committee

Nguyen Ngoc Thien An	University College Dublin, Ireland
Pedro Antunes	Victoria University of Wellington, New Zealand
Stephane Bressan	National University of Singapore, Singapore
Tran Cao De	Can Tho University, Vietnam
Thanh-Nghi Do	Can Tho University, Vietnam
Nguyen Van Doan	Japan Advanced Institute of Science and Technology, Japan
Dirk Draheim	University of Innsbruck, Austria
Patrick Etcheverry	University of Pau, France
Verena Geist	Software Competence Center Hagenberg, Austria
Raju Halder	Indian Institute of Technology Patna, India
Tran Van Hoai	HCMC University of Technology, Vietnam
Nguyen Viet Hung	University of Trento, Italy
Nguyen Quoc Viet Hung	The University of Queensland, Australia
Ryutaro Ichise	National Institute of Informatics, Japan
Tomohiko Igasaki	Kumamoto University, Japan
Koichiro Ishibashi	The University of Electro-Communications, Japan
Hiroshi Ishii	Tokai University, Japan
Kazuhiko Hamamoto	Tokai University, Japan
Abdelkader Hameurlain	Paul Sabatier University, Toulouse, France
Eiji Kamioka	Shibaura Institute of Technology, Japan

M-Tahar Kechadi	University College Dublin, Ireland
Nhien An Le Khac	University College Dublin, Ireland
Le Duy Khanh	Data Storage Institute, Singapore
Surin Kittitornkun	King Mongkut's Institute of Technology Ladkrabang, Thailand
Andrea Ko	Corvinus University of Budapest, Hungary
Lam Son Le	HCMC University of Technology, Vietnam
Faizal Mahananto	Institut Teknologi Sepuluh Nopember, Indonesia
Clavel Manuel	The Madrid Institute for Advanced Studies in Software Development Technologies, Spain
Christophe Marquesuzaa	University of Pau, France
Fabio Massacci	University of Trento, Italy
Atsuko Miyaji	Osaka University and Japan Advanced Institute of Science and Technology, Japan
Hiroaki Morino	Shibaura Institute of Technology, Japan
Nguyen Thai-Nghe	Cantho University, Vietnam
Thanh Binh Nguyen	HCMC University of Technology and Can Tho University of Technology, Vietnam
Benjamin Nguyen	Institut National des Sciences Appliqués Centre Val de Loire, France
An Khuong Nguyen	HCMC University of Technology, Vietnam
Khoa Nguyen	The Commonwealth Scientific and Industrial Research Organisation, Australia
Phan Trong Nhan	Johannes Kepler University Linz, Austria
Luong The Nhan	HCMC University of Technology, Vietnam
Eric Pardede	La Trobe University, Australia
Cong Duc Pham	University of Pau, France
Phu H. Phung	University of Dayton, USA
Tran Minh Quang	HCMC University of Technology, Vietnam
Le Thanh Sach	HCMC University of Technology, Vietnam
Tran Le Minh Sang	WorldQuant LLC, USA
Christin Seifert	University of Passau, Germany
Erik Sonnleitner	Johannes Kepler University Linz, Austria
Reinhard Stumptner	Software Competence Center Hagenberg, Austria
Tran Ngoc Thinh	HCMC University of Technology, Vietnam
Quoc Cuong To	German Research Center for Artificial Intelligent, Germany
Michel Toulouse	Vietnamese-German University, Vietnam
Huy Tran	University of Vienna, Austria
Ha-Manh Tran	International University, Vietnam
Le Hong Trang	Vinh University, Vietnam
Tuan Anh Truong	HCMC University of Technology, Vietnam and University of Trento, Italy
Tran Minh Triet	HCMC University of Natural Sciences, Vietnam
Nguyen Anh Tuan	University of Information Technology, VNUHCM, Vietnam

Truong Minh Nhat Quang	Can Tho University of Technology, Vietnam
Osamu Uchida	Tokai University, Japan
Hoang Tam Vo	IBM Research, Australia
Pham Tran Vu	HCMC University of Technology, Vietnam
Edgar Weippl	SBA Research, Austria

External Reviewers

Oyindamola Oluwatimi	Purdue University, USA
Thien Phan	National Institute of Informatics, Japan
Do Van Nguyen	Nagaoka University of Technology, Japan
Ai Thao Nguyen Thi	HCMC University of Technology, Vietnam
Quang Hai Truong	Data SecuriTy Applied Research Lab, Vietnam
Tran Tri Dang	Data SecuriTy Applied Research Lab, Vietnam
Bao Thu Le Thi	HCMC University of Technology, Vietnam

Contents

Big Data Analytics and Cloud Data Management

Internet of Things and Applications

Security and Privacy Engineering

Data Protection and Data Hiding

Advances in Authentication and Data Access Control

Access Control in NoSQL and Big Data

Context-Based Data Analysis and Applications

Emerging Data Management Systems and Applications

Big Data Analytics and Cloud Data Management

Incorporating Trust, Certainty and Importance of Information into Knowledge Processing Systems – An Approach

Markus Jäger(✉), Trong Nhan Phan, Christian Huber, and Josef Küng

Faculty of Engineering and Natural Sciences (TNF),
Institute for Application Oriented Knowledge Processing (FAW),
Johannes Kepler University (JKU), Linz, Austria
{mjaeger,nphan,chuber,jkueng}@faw.jku.at
http://www.faw.jku.at

Abstract. The origin of data (data provenance), should always be measured or categorized within the context of trusting the source of data. Can we be sure that the information we receive is trustworthy and reliable? Is the source trustable? Is the data certain? And how important is the received data the our current and next step of processing? We face these questions in the context of knowledge processing systems by developing a convenient approach to bring all these questions and values – trustability, certainty, importance – into a computable, measurable, and comparable way of expression. Not yet facing the question "How to compute trust or certainty?", but how to incorporate and process their measured values in knowledge processing systems to receive a representative view on the whole environment and its output.

Keywords: Trust · Certainty · Knowledge processing systems · Security · Risk · Provenance · Reliability

1 Introduction

When using a system, that processes data from external sources, a good quality of the data is not always ensured. It is important to know, how trustworthy a source is and how certain its data is. In our approach, we present a way to use and process values about trust of sources and certainty of data by taking into account different importances of different inputs. We do not cover the way, how trust and certainty are computed, we assume, that this step has been taken before. With these values, we continue processing and compute representative values of trust and certainty in knowledge processing systems.

The paper is structured as follows. In Sect. 2, we give some insight into related work and show definitions for trust, security, risk, and provenance in the common area. Section 3 presents our approach for incorporating trust and certainty into knowledge processing systems – covering several questions concerning this topic and giving a current view on our research. In Sect. 3.2, we transact the approach

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 3–19, 2016.
DOI: 10.1007/978-3-319-48057-2_1

into practice by defining the scopes of trust, certainty, and importance, and trying to give some options for further calculation to use these values in knowledge processing systems. In Sect. 4, we show examples of applying the approach on fictitious scenarios. The application of the approach in a real world scenario is shown in 4.3. Summing up in Sect. 5, we hope to give a complete view on our developed approach.

2 Definitions of Security, Risk, Trust and Provenance

2.1 Security

In our context, security mainly refers to computer security (protection of IT systems, information systems, protection of hard- and software, prevention of undesired intruders, etc.) and information security. "Security is the degree of protection against danger, damage, loss, and crime." [1].

The conclusion of our past researches and further researches (like [2]) is that cloud security cannot be established in the way as it should be or as we wish, because the responsibility of security and safety always is the business of the owner and provider of the cloud services. These are environment constraints which are unchangeable.

2.2 Risk

Risk in general addresses the potential of losing something with a special personal value. It is also seen as an intentional interaction with uncertainty, where the outcome is hard to predict [1].

Rousseau et al. [3] say, that *"Risk is the perceived probability of loss, as interpreted by a decision maker [...]. The path-dependent connection between trust and risk taking arises from a reciprocal relationship: risk creates an opportunity for trust, which leads to risk taking."*

Relating to information technology or information processing systems, risk can also be categorized as IT risk. This area of risk is a wide area of possible incidents, where a loss of values can occur in many different ways. This does not refer to the term uncertainty.

2.3 Trust

"In a social context, trust has several connotations. Definitions of trust typically refer to a situation characterized by the following aspects: One party (trustor) is willing to rely on the actions of another party (trustee); the situation is directed to the future. In addition, the trustor (voluntarily or forcedly) abandons control over the actions performed by the trustee. As a consequence, the trustor is uncertain about the outcome of the other's actions; they can only develop and evaluate expectations. The uncertainty involves the risk of failure or harm to the trustor if the trustee will not behave as desired." [1]

It always depends on the specific environment and field of research and application, what is understood by the term "Trust".

In one of our last publications, we raised an issue about trusting in technology, especially in smart home systems, whereas everybody's personal security and safeness can be touched in a very sensitive way [4].

The three main types of applicable trust after Rousseau et al. [3] are (1) trusting beliefs, (2) trusting intentions, and (3) trusting behaviours, where these three types are connected to each other: *"1. Trusting beliefs means a secure conviction that the other party has favorable attributes (such as benevolence, integrity, and competence), strong enough to create trusting intentions. 2. Trusting intentions means a secure, committed willingness to depend upon, or to become vulnerable to, the other party in specific ways, strong enough to create trusting behaviors. 3. Trusting behaviors means assured actions that demonstrate that one does in fact depend or rely upon the other party instead of on oneself or on controls. Trusting behavior is the action manifestation of willingness to depend. Each of these generic trust types can be applied to trust in IT. Trusting behavior-IT means that one securely depends or relies on the technology instead of trying to control the technology."*

Another point of view is the similarity of trusting people and trusting technology, especially information technology, where the main difference is within the application of trust in the specific area. We highly recommend reading the paper "Trust in Information Technology" from D. Harrison McKnight [5].

Another very interesting publication about trust in information sources is given by Hertzum et al. [6] in "Trust in information sources: seeking information from people, documents, and virtual agents". They compare the notion of trust between people and virtual agents, based on two empirical studies. The testimonials were software engineers and users of e-commerce systems. Some relational aspects concerning trust in the industrial marketing and management sector can be found in "Concerning trust and information" from Denize et al. [7].

A very good approach for measuring trust is given in "An Approach to Evaluate Data Trustworthiness Based on Data Provenance" [8].

2.4 Provenance

When we come into trust concerning trusting in data and trusting the sources of data, the term "Data Provenance" comes into account. It means the origin and complete processing history of any kind of data. A quite good introduction and overview can be found in "Data provenance – the foundation of data quality" [9] and in "Data Provenance: Some Basic Issues" [10]: *"We use the term data provenance to refer to the process of tracing and recording the origins of data and its movement between databases." and "It is an issue that is certainly broader than computer science, with legal and ethical aspects."*

Several problems concerning data provenance are covered in "Research Problems in Data Provenance" [11].

3 Developing a Convenient Approach

3.1 Introduction and Open Issues

We are currently developing a convenient approach for incorporating trust and certainty values into knowledge processing systems. The main principle is explained as follows. The main subjects in our approach are:

- any Source (S), which is providing information to an environment; there can be multiple sources in an environment;
- any Data (D)[1], which is provided by one Source – for our model, every Source usually provides one data, but also can provide more data;
- any Knowledge Processing System (KPS), which is processing data from one or more sources – each KPS itself is producing new data as output, in our model, every KPS produces only one output.

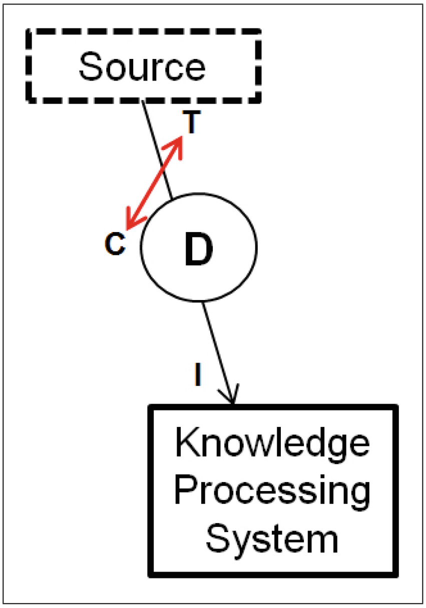


Fig. 1. Introduction to our approach.

The source is not yet specified more precise. It provides any data, not important which type of data – can be a whole database as well as a single text file or a data value. Details about the knowledge processing system are also not

[1] In our work, we combine the data and information layer, referring to the Data-Information-Knowledge-Wisdom (DIKW) architecture in [12] from Russell Lincoln Ackoff, so data has the role of information and belongs to the information layer.

important – it is any system using the provided data from the existing sources, processing the data and giving a new output of data. To have computable and usable values in our approach, computation of these different values from existing input data is needed. The main values in our approach are:

- Trust value (T) of source (S);
- Certainty value (C) of data (D);
- Importance value (I) of data (D), decided by the current knowledge processing system (KPS) for the current step of computation.

To go into further description of our approach, we have to introduce T, C and I and their relevance. Trust T is a value on how trustable your source is. You always have to see your system (sources / data / knowledge processing systems) as a whole environment, so the trust for a specific source should always be the same for this source.

Certainty C tells how reliable, confident or steady the provided data is. There exist many literature and research work about certainty and believability on knowledge based systems. We do not go into further handling of the questions "How to determine a value for trust T for a specific source?" or "How to determine a value for certainty C for specific data?". These questions have to be handled in further work.

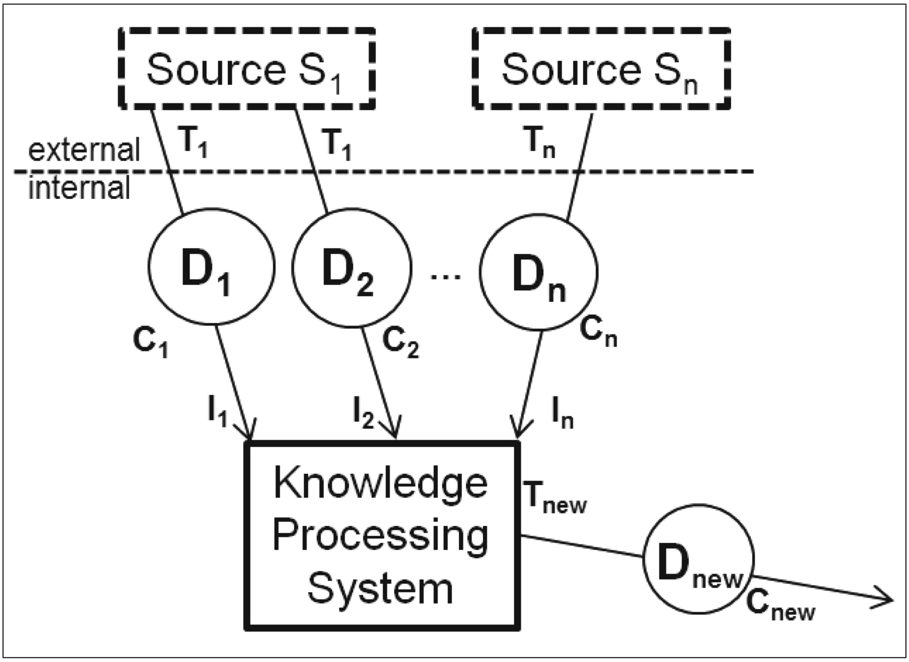


Fig. 2. Details on our approach.

The importance I belongs to the data but is defined by the KPS for every data separately – see Fig. 1. The reason for this is quite simple: in the KPS, some computations are made with all the input data and the system itself has to decide, which data is how important for it in this specific context (we assume that in every KPS, there is only one main usage of all the input respectively one computation which produces one output: another data D).

At this point, it is still open how the values for T, C, and I are defined and what their scopes will be. There is also unclear, how trust T is determined for a specific source and how the knowledge processing system decides on importance I for the data. To decide over certainty C, there are many approaches (as written before), but so far unclear is, how to put the certainty in a value for our new approach. There is also the question of what the scope of values will be and if there is normalization needed after calculation. For example, we propose values between 0 and 1 for certainty C and trust T, probably a three-step approximation for the importance I (e.g. 0.5 for unimportant, 1.0 for neutral, and 1.5 for important values). If the calculated new values (T_{new}, C_{new}) reach scopes beneath 1.0, the values have to be normalized for further usage (e.g. in a multiple-step system, where several knowledge processing systems are calculating T, C, and I values multiple times). We also assume, that C and T can be both dependent and independent, which needs to be defined – see the arrow in Fig. 1.

At the next step, we have to discuss about the continuation of processing T, C, and I. When there is a model of application or calculation, the new output value of C has to be re-applied on trust – because if the knowledge processing system generates an output which is used again, you have to consider a new trust value for this output. A non-trivial problem, because how do you measure or determine the trustworthiness of your knowledge processing system? Is it trustable because it is a known knowledge processing system in a controlled environment? If it is an internal part of the overall system, you might assume that you can trust this source, but (as seen in Fig. 2) the initial values can come from an external source, where the trustworthiness is not guaranteed.

There also arises the question of handling trust values in general for internal and external sources, and it has to be considered, that there should always be the same trust values for the same external/internal source – but it can be possible, that one source gains higher trust over time (e.g. when the values are continuous and recognized as stable and certain).

An approach for data provenance and measuring believablity of data that goes in a quite similar direction, but does not cover the usage in knowledge processing or knowledge based systems, can be found in [13].

In the next subsection, we transact the approach into practice by defining scopes and calculation and answering several of the arisen questions.

3.2 Scopes and Calculation

We are now making an attempt of concretizing the model described above by answering some of the questions and fixing the scopes of possible values as follows:

- Trust T of source S, for each S, has to be greater than 0 and less or equal than 1, where each value of T for each S has to be the same (if used multiple times) – a higher value represents higher trust:

$$0 < T \leq 1 \tag{1}$$

- Certainty C of data D, for each D, has to be greater than 0 and less or equal than 1, where each value of C for each D has to be the same (if used multiple times) – a higher value represents higher certainty:

$$0 < C \leq 1 \tag{2}$$

- Importance I of data D, decided by the KPS, is staggered:
  - 0.5 for values which are not very important;
  - 1.0 for regular values, where no special impact on importance is given;
  - 1.5 for very important values, concerning the current data processing.

$$I = 0.5 \mid 1 \mid 1.5 \tag{3}$$

Note: Regarding the current step of processing in the KPS for example: if data D_i is given the importance 1.5, there also has to be another data D_j with an importance of 0.5. So, there always have to be the same number of importance weighted D with 0.5 and with 1.5. The importance of 1.0, in fact, does not affect the current step of processing. This constraint guarantees avoiding an overestimation of grading input data too often as "very important", which would result in a deferral of representativeness of the output values.

Note: For this model it is necessary, that the input values of T and C are initialized like defined in (1) and (2). We do not give an answer on "How to calculate T and C?" - it is assumed, that this step has been taken before, (as written in Sect. 2, a very good approach for measuring trust is given in [8].

We now define the formulas for processing new T and C values as outcome of a KPS. This is the arithmetical average of the input T or C weighted with the current I for each D.

$$T_{new} = \frac{1}{n}\sum_{i=1}^{n}(T_i \times I_i) \tag{4}$$

Formula 4: Calculating T_{new} over all T_{1-n} related to I_{1-n}.

$$C_{new} = \frac{1}{n}\sum_{i=1}^{n}(C_i \times I_i) \tag{5}$$

Formula 5: Calculating C_{new} over all C_{1-n} related to I_{1-n}.

3.3 Alternative Aggregation Functions

In our research several other methods to calculate T$_{new}$ and C$_{new}$ have been discussed. For example the following functions are also possible for calculation: (Note: in the Eqs. 6–11, T$_{new}$ and T$_i$ can always be substituted with C$_{new}$ and C$_i$ to get the corresponding formulas, as our current intention is to compute T$_{new}$ and C$_{new}$ in the same manner).

$$T_{new} = \frac{1}{n}\sum_{i=1}^{n} T_i \tag{6}$$

$$T_{new} = \sum_{i=1}^{n}(T_i * I_i) \tag{7}$$

$$T_{new} = \frac{1}{n}\prod_{i=1}^{n}(T_i + I_i) \tag{8}$$

$$T_{new} = \frac{1}{n}\prod_{i=1}^{n} T_i \tag{9}$$

$$T_{new} = \prod_{i=1}^{n}(T_i + I_i) \tag{10}$$

$$T_{new} = \min_{n}(T_i) \tag{11}$$

For a first approach we use the aggregation functions 4 and 5 in the following sections, for now. The motivation for this is, that the decision of the finally used functions in the complete developed approach has not been taken.

Interpretation of general results: In our approach, there are currently no defined thresholds for the values. As the outcome of T$_{new}$ and C$_{new}$ can be seen as percentage-values, the interpretation of these values is free for every user. Of course, higher percentage means better trust and certainty.

4 Test-Scenarios

4.1 Simple Scenario

The following example relies on the provided model and formulas in Sect. 3.2. Introducing four sources (S$_1$ to S$_4$) with different trust values (T$_1$ to T$_4$), each providing one data (D$_1$ to D$_4$) with different certainty values (C$_1$ to C$_4$) for one knowledge processing system (KPS$_A$), which weights the different importances (I$_1$ to I$_4$). The values are listed in Table 1 as follows:

The application of the formulas 4 and 5 are leading to the following results:

$$T_{new} = \frac{0.8 \times 1 + 0.4 \times 1.5 + 0.9 \times 0.5 + 0.2 \times 1}{4} = \frac{2.05}{4} = 0.5125 \tag{12}$$

$$C_{new} = \frac{0.9 \times 1 + 0.2 \times 1.5 + 0.2 \times 0.5 + 0.7 \times 1}{4} = \frac{2}{4} = 0.5 \tag{13}$$

Table 1. Initial values of T, C and I for a simple scenario.

S_1: $T_1 = 0.8$	D_1: $C_1 = 0.9$	KPS_A: $I_1 = 1.0$
S_2: $T_2 = 0.4$	D_2: $C_2 = 0.2$	KPS_A: $I_2 = 1.5$
S_3: $T_3 = 0.9$	D_3: $C_3 = 0.2$	KPS_A: $I_3 = 0.5$
S_4: $T_4 = 0.2$	D_4: $C_4 = 0.7$	KPS_A: $I_4 = 1.0$

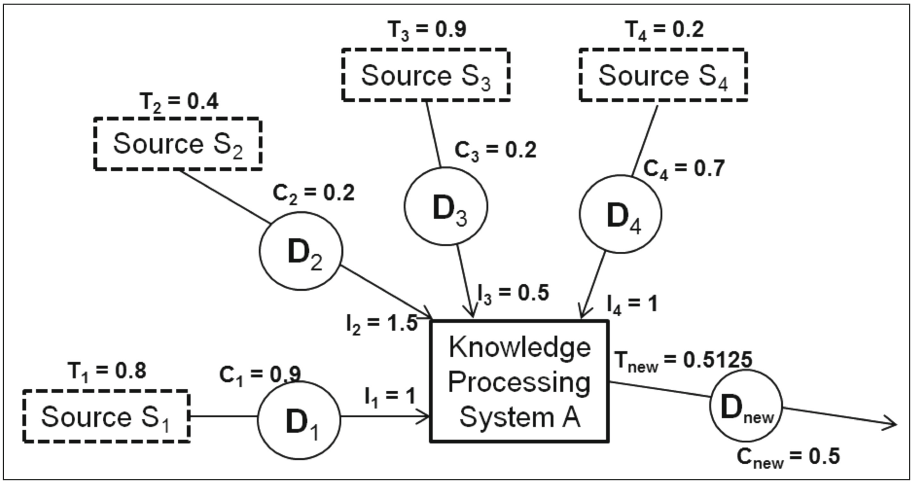


Fig. 3. Simple Test Scenario.

Evaluation of Simple Scenario: As seen in Fig. 3 and in the calculation above, the outcome of T_{new} and C_{new} are very representative, regarding the input values, which were chosen well balanced (the mixture of T and C values was chosen in a way, where all possible situations are represented with four inputs: high/low T with high/low C and each reversed).

To describe the model in detail: the most impact on the final score is given through S_2 because of its weighting by $I_2 = 1.5$ – both T and C values from this source are very low (0.4 and 0.2), which affects the final score in a meaningful way. The highest trust is provided by S_3, but because of its low importance in the current processing step, it does not affect the result that much (in fact, only 1/3 as hard as S_2 does). The remaining trust values from S_1 and S_3 are quite an average of impact, because their weighted value is 0.5. The same situation counts for the certainty values C in this scenario. This is the reason for the very mean outcome of $T_{new} = 0.5125$ and $C_{new} = 0.5$.

We know, that an application of our model in such a small use case shown here is only the representation of simple reality. Most of the time, various processing steps occur in multiple knowledge processing systems. We go into a more realistic application in the next subsection.

4.2 Advanced Scenario

For our advanced scenario, we are introducing six sources (S_1 to S_6) with different trust values (T_1 to T_6), each providing one or two data (D_{11}, D_{12}, D_2, D_{31}, D_{32}, D_4, D_{51}, D_{52}, and D_6) with different certainty values (C_{11}, C_{12}, C_2, C_{31}, C_{32}, C_4, C_{51}, C_{52}, and C_6) for multiple knowledge processing systems (KPS_A to KPS_E), which weight the different importance.

KPS_{A-C} are working only with Data from sources S_{1-6}, so the T and C values are given. KPS_D is working with Data from Source S_1 and $KPS_{A,B}$ and KPS_E is processing only output values from $KPS_{B,C,D}$ (no initial sources) – so the calculation of T and C of KPS_D and KPS_E depend on the calculations of $KPS_{A,B,C}$, because they receive (most of) their input T and C values from previous processing steps. An overview over the advanced scenario including the calculated values is shown in Fig. 4 at the end of this section. The values for the first calculation step are listed in Table 2 as follows:

Table 2. Initial values of T, C & I for an advanced scenario.

S_1: $T_1 = 1.0$	D_{11}: $C_{11} = 0.9$	KPS_A: $I_{A1} = 0.5$
S_2: $T_2 = 0.4$	D_2: $C_2 = 0.3$	KPS_A: $I_{A2} = 1.0$
S_3: $T_3 = 0.8$	D_{31}: $C_{31} = 0.8$	KPS_A: $I_{A3} = 1.5$
	D_{32}: $C_{32} = 0.5$	KPS_B: $I_{B1} = 1.0$
S_4: $T_4 = 0.2$	D_4: $C_4 = 0.2$	KPS_B: $I_{B2} = 1.5$
S_5: $T_5 = 0.5$	D_{51}: $C_{51} = 1.0$	KPS_B: $I_{B3} = 0.5$
	D_{52}: $C_{52} = 0.7$	KPS_C: $I_{C1} = 1.0$
S_6: $T_6 = 0.9$	D_6: $C_6 = 1.0$	KPS_C: $I_{C2} = 1.0$

Because of the fictional character of this scenario (with no detailed information about the involved knowledge processing systems), all values are chosen freely – especially the importance values are picked in a way to show the practicability as much as possible.

With these input values, we are able to compute trust and certainty values for the output data of KPS_{A-C} in a first step with the formulas 4 and 5, similar to the simple scenario in the previous section:

$$T_A = \frac{1 \times 0.5 + 0.4 \times 1 + 0.8 \times 1.5}{3} = \frac{2.1}{3} = 0.7 \tag{14}$$

$$C_A = \frac{0.9 \times 0.5 + 0.3 \times 1 + 0.8 \times 1.5}{3} = \frac{1.95}{3} = 0.65 \tag{15}$$

$$T_B = \frac{0.8 \times 1 + 0.2 \times 1.5 + 0.5 \times 0.5}{3} = \frac{1.35}{3} = 0.45 \tag{16}$$

$$C_B = \frac{0.5 \times 1 + 0.2 \times 1.5 + 1 \times 0.5}{3} = \frac{1.05}{3} = 0.35 \tag{17}$$

$$T_C = \frac{0.5 \times 1 + 0.9 \times 1}{2} = \frac{1.4}{2} = 0.7 \tag{18}$$

$$C_C = \frac{0.7 \times 1 + 1 \times 1}{2} = \frac{1.7}{2} = 0.85 \tag{19}$$

To give a structured way of progress, we accumulate the calculated values in Table 3, for the ongoing process of calculating the output of KPS$_D$ and KPS$_E$. Note, that KPS$_A$, KPS$_B$, and KPS$_C$ act as new/additional sources for the ongoing calculations.

Table 3. Calculated values and initial values for processing output of KPS$_D$.

S$_1$: T$_1 = 1.0$	D$_{12}$: C$_{12} = 0.80$	KPS$_D$: I$_{D1} = 0.5$
KPS$_A$: T$_A = 0.7$	D$_A$: C$_A = 0.65$	KPS$_D$: I$_{D2} = 1.0$
KPS$_B$: T$_B = 0.45$	D$_B$: C$_B = 0.35$	KPS$_D$: I$_{D3} = 1.5$

With these calculated values, we can now progress calculating the scenario by computing the values for KPS$_D$.

$$T_D = \frac{1 \times 0.5 + 0.7 \times 1 + 0.45 \times 1.5}{3} = \frac{1.875}{3} = 0.625 \tag{20}$$

$$C_D = \frac{0.8 \times 0.5 + 0.65 \times 1 + 0.35 \times 1.5}{3} = \frac{1.575}{3} = 0.525 \tag{21}$$

The important values for calculating the output of KPS$_E$ are shown in Table 4:

Table 4. Values for final processing step.

KPS$_C$: T$_C = 0.70$	D$_C$: C$_C = 0.85$	KPS$_E$: I$_{E1} = 1.0$
KPS$_D$: T$_D = 0.625$	D$_D$: C$_D = 0.525$	KPS$_E$: I$_{E2} = 1.0$

With these calculated values, we can now progress finishing the scenario by computing the values for KPS$_D$ and KPS$_E$, where KPS$_E$ generates the final output values of this scenario.

$$T_E = \frac{0.625 \times 1 + 0.7 \times 1}{2} = \frac{1.325}{2} = 0.6625 \tag{22}$$

$$C_E = \frac{0.525 \times 1 + 0.85 \times 1}{2} = \frac{1.375}{2} = 0.6875 \tag{23}$$

The results of this advanced scenario are:

- Trust T$_E$ of KPS$_E$ is computed with 0.6625
- Certainty C$_E$ of D$_E$ is computed with 0.6875

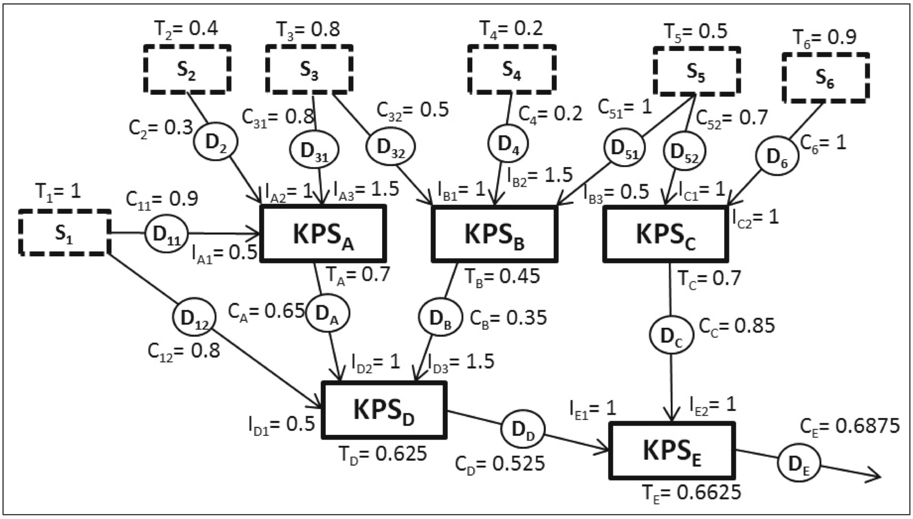


Fig. 4. Advanced Test Scenario.

Evaluation of Advanced Scenario. Here, the highest impact in the whole calculation concerning trust the sources S_3 and S_4 have because of their high importance in KPS_A and KPS_B as well as in the further calculation step in KPS_D. Concerning certainty, the low value of $C_4 = 0.2$ takes into account in this model, as its importance is rated with 1.5.

If you represent the final results in a normalized way, you can interpret them with $T_E = 66.25\,\%$ and $C_E = 68.75\,\%$ which can be seen as a good representational view on the whole systems' trust and certainty outcome, similar representative as the outcome in the simple scenario.

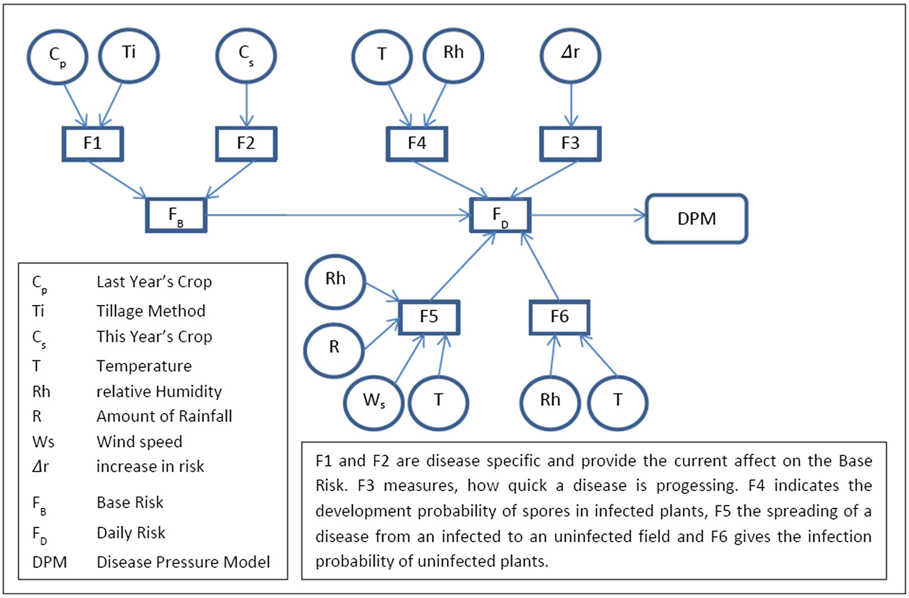
4.3 Application in a Real World Scenario

In this subsection, we present the application of our approach on a real world scenario, which we used in our current research- and project work.

Disease Pressure Model. We are now referring to the DPM (Disease Pressure Model, used in the Project CLAFIS [14]) for calculating an accurate (daily) risk value, how certain a specific disease outbreak for a specific field can be. The DPM is sketched in Fig. 5, where also the explanation of the single parts is given, as well as the description of the used functions.

The DPM uses input values from a FMIS (farm management information system) where information like this year's crop, last year's crop and the used tillage method are stored. The needed weather data comes from (several) weather stations, where information like temperature, relative humidity, amount of rainfall and wind speed is gathered.

We chose the DPM because it is used in a current project, we are working on.



C_p	Last Year's Crop
Ti	Tillage Method
C_s	This Year's Crop
T	Temperature
Rh	relative Humidity
R	Amount of Rainfall
Ws	Wind speed
Δr	increase in risk
F_B	Base Risk
F_D	Daily Risk
DPM	Disease Pressure Model

F1 and F2 are disease specific and provide the current affect on the Base Risk. F3 measures, how quick a disease is progessing. F4 indicates the development probability of spores in infected plants, F5 the spreading of a disease from an infected to an uninfected field and F6 gives the infection probability of uninfected plants.

Fig. 5. Disease Pressure Model (DPM) – sketched.

Application of Approach on DPM. We now apply our approach on the model of DPM. The whole process of calculation can be seen in Fig. 6, the initial values, which are intentional an fictitious, for the first steps of calculation are listed in Table 5.

Table 5. Initial values of T, C and I for calculating F1, F2, F4, F5, and F6.

$T_{FMIS} = 1$	$C_{Cp} = 0.8$ $C_{Ti} = 0.9$ $C_{Cs} = 0.8$	F1: $I_{F1-Cp} = 1$ F1: $I_{F1-Ti} = 1$ F2: $I_{F2-Cs} = 1$
$T_{WeatherStation1} = 0.9$	$C_T = 0.9$	F4: $I_{F4-T} = 0.5$ F6: $I_{F6-T} = 0.5$ F5: $I_{F5-T} = 1$
	$C_{Rh} = 0.8$	F4: $I_{F4-Rh} = 1.5$ F6: $I_{F6-Rh} = 0.5$ F5: $I_{F5-Rh} = 1$
$T_{WeatherStation2} = 0.7$	$C_R = 0.7$ $C_{Ws} = 0.8$	F5: $I_{F5-R} = 0.5$ F4: $I_{F5-Ws} = 1.5$

The application of the formulas 4 and 5 are leading to the following results of trust (T_{F1}, T_{F2}, T_{F4}, T_{F5}, T_{F6}) and certainty (C_{DF1}, C_{DF2}, C_{DF4}, C_{DF5}, C_{DF6}):

$$T_{F1} = 1 \quad T_{F2} = 1 \tag{24}$$

$$C_{DF1} = 0.85 \quad C_{DF2} = 0.8 \tag{25}$$

$$T_{F4} = 0.9 \quad T_{F6} = 0.9 \quad T_{F5} = 0.8 \tag{26}$$

$$C_{DF4} = 0.825 \quad C_{DF6} = 0.875 \quad C_{DF5} = 0.8125 \tag{27}$$

As a step in between, we have to calculate the outcome of FBase, to be able to process the final calculations. The needed input values for FBase are listed in Table 6.

Table 6. Values for calculating the outcome of FBase.

$\mathrm{T}_{F1} = 1$	$\mathrm{C}_{DF1} = 0.85$	FB: $\mathrm{I}_{FB-DF1} = 1$
$\mathrm{T}_{F2} = 1$	$\mathrm{C}_{DF2} = 0.8$	FB: $\mathrm{I}_{FB-DF2} = 1$

$$T_{FB} = 1 \qquad C_{DFB} = \frac{0.85 + 0.8}{2} = 0.825 \tag{28}$$

With the calculation of trust and certainty of FBase, F4, F6, and F5, we are now able to continue the scenario.

Table 7. Values for calculating the outcome of FDaily.

$\mathrm{T}_{FB} = 1$	$\mathrm{C}_{DFB} = 0.825$	FD: $\mathrm{I}_{FD-DFB} = 1.5$
$\mathrm{T}_{F4} = 0.9$	$\mathrm{C}_{DF4} = 0.825$	FD: $\mathrm{I}_{FD-DF4} = 1$
$\mathrm{T}_{F6} = 0.9$	$\mathrm{C}_{DF6} = 0.875$	FD: $\mathrm{I}_{FD-DF6} = 1$
$\mathrm{T}_{F5} = 0.8$	$\mathrm{C}_{DF5} = 0.8125$	FD: $\mathrm{I}_{FD-DF5} = 0.5$

With the calculated values, we now can finish the scenario by measuring the outcome of trust T_{FD} and certainty C_{DFD} of the function FDaily. The needed values for calculating FDaily are listed in Table 7.

$$T_{FD} = \frac{1 \times 1.5 + 0.9 \times 1 + 0.9 \times 1 + 0.5 \times 0.8}{4} = \frac{3.7}{4} = \underline{\underline{0.925}} \tag{29}$$

$$\begin{aligned} C_{DFD} &= \frac{0.825 \times 1.5 + 0.825 \times 1 + 0.875 \times 1 + 0.8 \times 0.5}{4} \\ &= \frac{3.34375}{4} = 0.8359375 = \underline{\underline{0.836}} \end{aligned} \tag{30}$$

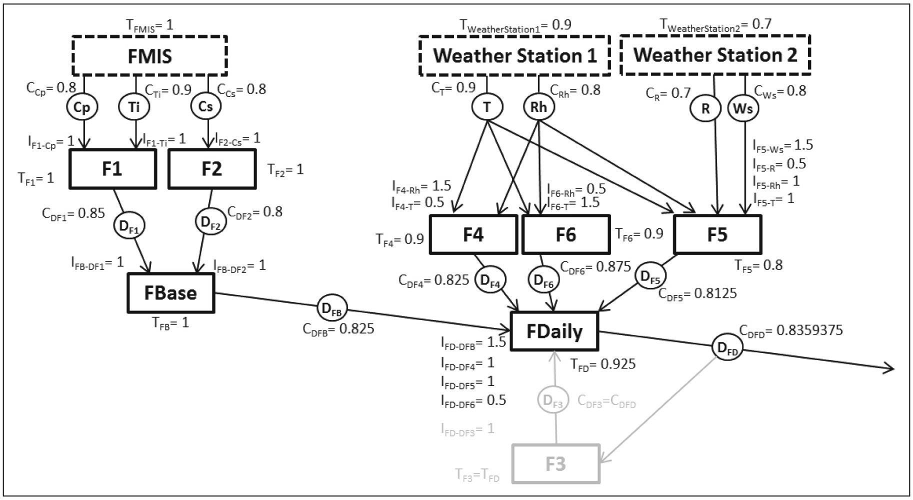


Fig. 6. Approach with DPM.

Evaluation of DPM. The DPM model is a quite satisfying scenario for our approach, because the sources are "under our control", which means, that these sources are highly trustable ($\mathrm{T}_{FMIS} = 1$, $\mathrm{T}_{WeatherStation1} = 0.9$, $\mathrm{T}_{WeatherStation2} = 0.7$ – well, the chosen trust value of $\mathrm{T}_{WeatherStation2}$ is for a better variation in our model – in fact, in real it would be similar high as $\mathrm{T}_{WeatherStation1}$) and the provided data can convince with a high certainty, because the values in the FMIS are entered by the person who controls the FMIS (most of the time the farmer himself) and the provided data from the weather stations should be very accurate.

The most important step in the calculation of the DPM is the providing of the base risk, where the outcome of FBase is very good for further calculations: T_{FB} stays on the highest possible value with 1 and the certainty of $\mathrm{C}_{DFB} = 0.825$ is also a very good indicator for FDaily. Particularly because of the importance of FBase in FDaily $\mathrm{I}_{FD-DFB} = 1.5$, its affection is quite high on the last step of calculation.

With the outcome of FDaily with $\mathrm{T}_{FD} = 0.925$ and the certainty $\mathrm{C}_{DFD} = 0.836$ of FDaily's produced output D_{FD}, we can sum up, that the DPM is producing quite trustable and certain values. If the values should be represented in a normalized way, you can interpret $\mathrm{T}_{FD} = 92.5\,\%$ and $\mathrm{C}_{DFD} = 83.6\,\%$ which are quite high and good values for trust and certainty in a system, where these values are important.

Note: as seen in Fig. 6, there is an area in grey: the values of function F3, which are the same as from FDaily would come into account as an input for the calculation of FDaily itself. As recursion is not yet covered in our approach, we do not consider the function F3 in this work and in the calculation of trust and certainty values in the DPM, but there will be a consideration of recursion in further research work by improving our approach.

5 Summary

We addressed the question of how to determine trust- and certainty-values of an output of a KPS, when different trust- and certainty-values for the input data are given.

We showed a first solution on a simple and advanced example as well as on a real world scenario, the disease pressure model (DPM). The results are realistic and the computed values are promising.

Further steps like analyzing runtime-complexity, proof of non-converging, evaluation of usage of the approach, experiments and testing the approach on several more realistic multi-step scenarios and their evaluation will be done in further work, as well as the evaluation of more complex aggregation functions, probably incorporating statistical distributions of trust and certainty values. Also the step of considering recursion in the approach will be observed, and the facing of questions like “Is staggering of I needed?” and “Are T and C (in)dependent?”.

A philosophical element which has to be discussed is the following: “Is it in principle allowed to alter a trust value according to its importance?” Interpreting and naming it as influence probably would make less problems in this aspect.

Our aim is to develop an overall approach for incorporating trust-, certainty- and importance/influence values to gain a complete model for calculating representative values in knowledge processing systems. The approach then can also be applied to all other types of processing systems.

Acknowledgments. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no.604659.

The authors would like to thank all partners in CLAFIS [14] for the good cooperation within the project. The collaboration between experts from different fields made it possible, to write a paper with this content. Special thanks to our colleagues from LUKE in Finland for the development of the DPM (disease pressure model) to give a possibility, to apply the approach on a practical and real world scenario in our current project-work.

References

1. Online: Wikipedia, the free Encyclopedia (2015)
2. Christodorescu, M., Sailer, R., Schales, D.L., Sgandurra, D., Zamboni, D.: Cloud security is not (just) virtualization security: a short paper. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security (2009)
3. Rousseau, D., Sitkin, S., Burt, R., Camerer, C.: Not so different after all: a cross-discipline view of trust. Acad. Manage. Rev. **23**(3), 393–404 (1998)
4. Jäger, M., Nadschläger, S., Nhan Phan, T.: Towards the trustworthiness of data, information, knowledge and knowledge processing systems in smart homes. In: IDIMT-2015 Information Technology and Society Interaction and Interdependence, vol. 23. Trauner, September 2015. http://www.idimt.org, ISBN:978-3-99033-395-2

5. McKnight, D.H.: Trust in Information Technology. Blackwell Encycl. Manage. Oper. Manage. **7**, 329–331 (2005). Blackwell Pub
6. Hertzum, M., Andersen, H.H., Andersen, V., Hansen, C.B.: Trust in information sources: seeking information from people, documents, and virtual agents. Interact. Comput. **14**(5), 575–599 (2002)
7. Denize, S., Young, L.: Concerning trust and information. Ind. Mark. Manage. **36**(7), 968–982 (2007). Opening the network - Bridging the IMP tradition and other research perspectives2006 IMP Conference Special Issue22nd Industrial Marketing and Purchasing Group Conference
8. Dai, C., Lin, D., Bertino, E., Kantarcioglu, M.: An approach to evaluate data trustworthiness based on data provenance. In: Jonker, W., Petković, M. (eds.) SDM 2008. LNCS, vol. 5159, pp. 82–98. Springer, Heidelberg (2008)
9. Buneman, P., Davidson, S.B.: Data provenance-the foundation of data quality (2010)
10. Buneman, P., Khanna, S., Tan, W.-C.: Data provenance: some basic issues. In: Kapoor, S., Prasad, S. (eds.) FSTTCS 2000. LNCS, vol. 1974, pp. 87–93. Springer, Heidelberg (2000). doi:10.1007/3-540-44450-5_6
11. Tan, W.C.: Research problems in data provenance. IEEE Data Eng. Bull. **27**, 45–52 (2004)
12. Ackoff, R.L.: From data to wisdom. J. Appl. Syst. Anal. **16**(1), 3–9 (1989)
13. Prat, N., Madnick, S.: Measuring data believability: a provenance approach. In: Hawaii International Conference on System Sciences, p. 393 (2008)
14. European Commission: http://www.clafis-project.eu/ y.n.: CLAFIS: Crop, livestock and forests integrated system for intelligent automation

Incremental Parallel Support Vector Machines for Classifying Large-Scale Multi-class Image Datasets

Thanh-Nghi Do[1,2](✉) and Minh-Thu Tran-Nguyen[1]

[1] College of Information Technology, Can Tho University, Can Tho 92100, Vietnam
{dtnghi,tnmthu}@cit.ctu.edu.vn
[2] UMI UMMISCO 209 (IRD/UPMC), Can Tho, Vietnam

Abstract. In this paper, we propose an incremental parallel support vector machines (SVM) training with stochastic gradient descent (SGD) for dealing with the very large number of images and large-scale multi-class on standard personal computers (PCs). The two-class SVM-SGD algorithm is extended in several ways to develop the new incremental parallel multi-class SVM-SGD in large-scale classifications. We propose the balanced batch SGD of SVM (BBatch-SVM-SGD) for trainning two-class classifiers used in the one-versus-all strategy of the multi-class problems and the incremental training process of classifiers in parallel way on multi-core computers. The numerical test results on ImageNet datasets show that our algorithm is efficient compared to the state-of-the-art linear SVM classifiers in terms of training time, correctness and memory requirements.

Keywords: Large-scale multi-class image classification · Incremental training · Support vector machines (SVM) · Stochastic gradient descent (SGD)

1 Introduction

The classification of images is one of the most important research topics in computer vision and machine learning. The purpose of the image classification is to automatically assign predefined categories to images. Its applications include handwriting character recognition, zip code recognition for postal mail sorting, numeric entries in forms filled up by hand, fingerprint recognition, face recognition, auto-tagging images and so on. The image classification task involves the main steps as follows: extracting features and building code-book, training classifiers. The popular systems of image classification (first publications [1,2]) for representing images use the Scale-Invariant Feature Transform method (SIFT [3,4]), the Bag-of-visual-Words representation model (BoW). The SIFT algorithm is to detect and describe local features in images which are invariant to image scale, rotation and also robust to changes in illumination, noise, occlusion. And then, k-means algorithm [5] performs the clustering task on descriptors to

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 20–39, 2016.
DOI: 10.1007/978-3-319-48057-2_2

form visual words from the local descriptors. The representation of the image for classification is the bag-of-words is constructed from the counting of the occurrence of words in a histogram like fashion. The step of the feature extraction and the BoW representation leads to datasets with very large number of dimensions (e.g. thousands of dimensions). The SVM algorithms [6] are suited for dealing with very-high-dimensional datasets. In spite of the accurate classification models, SVMs are not favorable to handle the challenge of large datasets. SVM solutions are obtained from quadratic programming, so that the computational cost [7] is at least square of the number of training datapoints and the memory requirement making SVM impractical.

The effective heuristics to improve SVM learning task are to divide the original QP into series of small problems [7,8]. Incremental learning [9–12] try to update solutions in growing training set. The other techniques include parallel and distributed learning on PC network [13,14], on graphics processing units [15] or choosing active set [16–18] for learning, ensemble-based [19], local learning [20–22], using the stochastic gradient descent for large scale linear SVM solvers [23–27]. However, these proposed algorithms are difficult to deal with large-scale multi-class image datasets on PCs (e.g. Caltech with 101 classes [28], Caltech with 256 classes [29] having hundreds of classes, and ImageNet dataset [30] with more than 14 million images in 21,841 classes). It yields huge classification challenges of very-high-dimensional and large-scale multi-class image datasets. For scaling-up the training in practice, the data is first transformed by a nonlinear mapping induced by a particular kernel and then the efficient linear classifiers are trained on the mapping space [30]. Furthermore, the comparative study in [31] shows that the training of a linear SVM is about 600 times faster than the training of a non-linear one with the same accuracy.

This challenge motivates us to study an efficient incremental linear SVM training for dealing with the very large number of images and large-scale multi-class on standard personal computers (PCs). We propose the extensions of the stochastic gradient descend (SGD [23,24]) for two-class SVM to develop the new incremental parallel multi-class SVM-SGD for efficiently classifying large image datasets into many classes. Our contributions include:

1. the balanced batch stochastic gradient descend of support vector machine (BBatch-SVM-SGD) for very large number of classes,
2. the incremental training process of classifiers in parallel way on multi-core computers.

The numerical test results on ImageNet datasets [30] show that our algorithm is efficient compared to the state-of-the-art linear SVM classifiers in terms of training time, correctness and memory requirements.

The remainder of this paper is organized as follows. Section 2 briefly presents the SGD algorithm for two-class SVM problems. Section 3 describes how to extend the two-class SVM-SGD to develop the new incremental parallel multi-class SVM-SGD for efficiently classifying large image datasets into many classes. Section 4 presents evaluation results, before the conclusions and future work in Sect. 5.

2 Stochastic Gradient Descent for Binary Classification of Support Vector Machines

2.1 Support Vector Machines for Binary Classification

Let us consider a linear binary classification task, as depicted in Fig. 1. The dataset D consists of m datapoints $\{x_1, x_2, \ldots, x_m\}$ in the n-dimensional input space R^n, having corresponding labels $\{y_1, y_2, \ldots, y_m\}$ being ± 1. For this classification problem, the SVM algorithms [6] try to find the best separating plane (denoted by the normal vector $w \in R^n$ and the scalar $b \in R$), i.e. furthest from both class $+1$ and class -1. It is accomplished through the maximization of the margin (or the distance) between the supporting planes for each class ($x.w - b = +1$ for class $+1$, $x.w - b = -1$ for class -1). The margin between these supporting planes is $2/\|w\|$ (where $\|w\|$ is the 2-norm of the vector w). Any point x_i falling on the wrong side of its supporting plane is considered to be an error, its error distance denoted by $z_i \geq 0$. Therefore, SVM has to simultaneously maximize the margin and minimize the error. The standard SVM pursues these goals with the quadratic programming (1).

$$\begin{aligned} \min\ \Psi(w, b, z) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} z_i \\ s.t.: y_i(w.x_i - b) + z_i \geq 1 \\ z_i \geq 0 \end{aligned} \tag{1}$$

where the positive constant C is used to tune errors and margin size.

The plane (w, b) is obtained by solving the quadratic programming (1). Then, the classification of a new datapoint x based on the plane is:

$$predict(x) = sign(w.x - b) \tag{2}$$

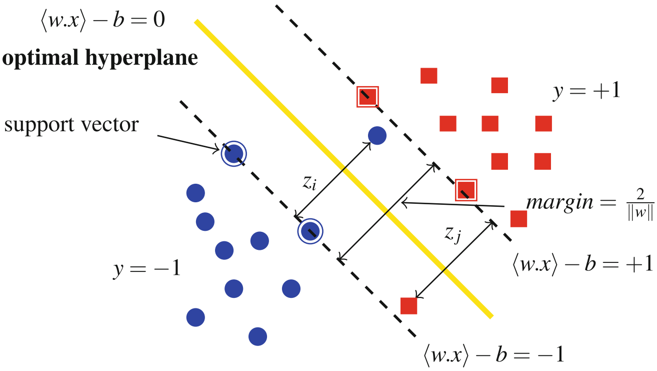

$\langle w.x \rangle - b = 0$
optimal hyperplane
support vector
$y = +1$
z_i
$margin = \frac{2}{\|w\|}$
z_j
$y = -1$
$\langle w.x \rangle - b = +1$
$\langle w.x \rangle - b = -1$

Fig. 1. Linear separation of the datapoints into two classes

SVM can use some kernel functions (e.g. a polynomial function of degree d or a Radial Basis Function) for dealing with non-linear classification tasks. More details about SVM and others kernel-based learning methods can be found in [32].

The study in [7] illustrated that the computational cost requirements of the SVM solutions in (1) are at least $O(m^2)$ (where m is the number of training datapoints), making standard SVM intractable for large datasets.

2.2 Stochastic Gradient Descent for Binary Classification of SVM

We can reformulate the SVM problem in quadratic programming (1) in an unconstraint problem. We can ignore the bias b without generality loss. The constraints $y_i(w.x_i) + z_i \geq 1$ in (1) are rewritten as follows:

$$z_i \geq 1 - y_i(w.x_i) \tag{3}$$

The constraints (3) and $z_i \geq 0$ are rewritten by the hinge loss function:

$$z_i = max\{0, 1 - y_i(w.x_i)\} = L(w, [x_i, y_i]) \tag{4}$$

Substituting for $z_i = L(w, [x_i, y_i])$ from the constraint in terms of w into the objective function Ψ of the quadratic programming (1) yields an unconstrained problem (5):

$$\min\ \Psi(w, [x, y]) = \frac{\lambda}{2}\|w\|^2 + \frac{1}{m}\sum_{i=1}^{m} L(w, [x_i, y_i]) \tag{5}$$

And then, [23,24] proposed the stochastic gradient descent method to solve the unconstrained problem (5). The stochastic gradient descent for SVM (denoted by SVM-SGD) updates w on T epochs with a learning rate η. For each epoch t, the SVM-SGD uses a single randomly received datapoint (x_i, y_i) to compute the sub-gradient $\nabla_t \Psi(w, [x_i, y_i])$ and update w_{t+1} as follows:

$$w_{t+1} = w_t - \eta_t \nabla_t \Psi(w, [x_t, y_t]) = w_t - \eta_t(\lambda w_t + \nabla_t L(w, [x_t, y_t])) \tag{6}$$

$$\nabla_t L(w, [x_t, y_t]) = \nabla_t max\{0, 1 - y_t(w.x_t)\} = \begin{cases} -y_t x_t & if\ y_t(w.x_t) < 1 \\ 0 & otherwise \end{cases} \tag{7}$$

The SVM-SGD using the update rule (6) is described in Algorithm 1.

As mentioned in [23,24], the SVM-SGD algorithm quickly converges to the optimal solution due to the fact that the unconstrained problem (5) is convex optimization problems on very large datasets. The algorithmic complexity of SVM-SGD is linear with the number of datapoints. An example of its effectiveness is given with the classification into two classes of 780 000 datapoints in 470000-dimensional input space in 2 s on a PC and the test accuracy is similar to standard SVM.

Algorithm 1. SVM-SGD algorithm for binary classification

```
input :
        training dataset $D$
        positive constant $\lambda > 0$
        number of epochs $T$
output:
        hyperplane $w$
begin
    init $w_1 = 0$
    for $t \leftarrow 1$ to $T$ do
        randomly pick a datapoint $[x_t, y_t]$ from training dataset $D$
        set $\eta_t = \frac{1}{\lambda t}$
        if $(y_i(w_t.x_i) < 1)$ then
            $w_{t+1} = w_t - \eta_t(\lambda w_t - y_i x_i)$
        else
            $w_{t+1} = w_t - \eta_t \lambda w_t$
        end
    end
    return $w_{t+1}$
end
```

3 Incremental Parallel SVM-SGD for Large-Scale Multi-class

The original SVM algorithms are only able to deal with two-class problems. There are several extensions of a two-class SVM solver for multi-class (c classes, $c \geq 3$) classification tasks. The state-of-the-art multi-class SVMs are categorized into two types of approaches. The first one is to consider the multi-class problem in an optimization problem [33–35]. The second one is to decompose multi-class into a series of binary SVMs, including one-versus-all [6], one-versus-one [36], Decision Directed Acyclic Graph and hierarchical methods for multi-class SVM [37–39] (hierarchically partitioning the data into two subsets).

In practice, the most popular methods are One-Versus-All (ref. LIBLINEAR [40]), One-Versus-One (ref. LibSVM [41]) and are due to their simplicity. The One-Versus-All strategy builds c different binary SVM models where the i^{th} one separates the i^{th} class from the rest, illustrated in Fig. 2. The One-Versus-One strategy constructs $c(c1)/2$ binary SVM models for all the binary pairwise combinations of the c classes, illustrated in Fig. 3. The class is then predicted with the largest distance vote.

When dealing with very large number of classes, e.g. $c = 1,000$ classes, the one-versus-one strategy is too expensive because it needs training $499,500$ of binary classifiers and using them in the classification (compared to $1,000$ binary models learned by the one-versus-all strategy). Therefore, the one-versus-all strategy is suited for this case. And then, our multi-class SVM-SGD algorithm also use the one-versus-all approach to train independently c binary classifiers.

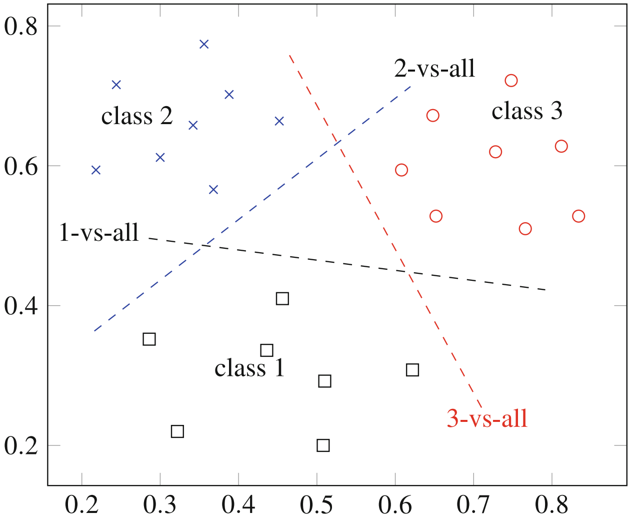


Fig. 2. Multi-class SVM (One-Versus-All)

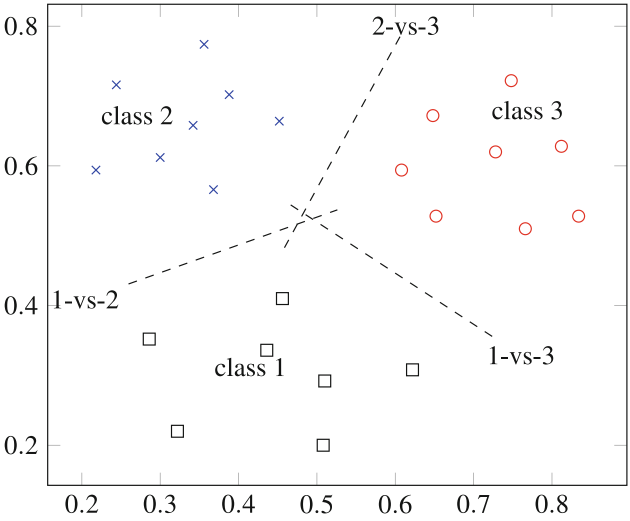


Fig. 3. Multi-class SVM (One-Versus-One)

Therefore, the multi-class SVM-SGD algorithm using one-versus-all leads to the two problems:

1. the SVM-SGD algorithm deals with the imbalanced datasets for building binary classifiers,

2. the SVM-SGD algorithm also takes very long time to train very large number of binary classifiers in sequential mode using a single processor,
3. furthermore, loading the whole large training dataset into memory requires very large memory capacity.

Due to these problems, we propose three ways for creating the new incremental parallel multi-class SVM-SGD algorithm (denoted by Incr-Par-MC-SVM-SGD) being able to handle the very large number of datapoints and large-scale multi-class on standard personal computers (PCs) in the high speed. The first one is to build balanced batch binary classifiers with under-sampling strategy. The second one is to parallelize the training task of all binary classifiers with several multi-core machines. The last one is the incremental learning of parallel multi-class SVM-SGD that avoids loading the whole large training dataset into memory.

3.1 Balanced Batch of Binary SVM-SGD Classifier

In the one-versus-all approach, the learning task of binary SVM-SGD classifier is try to separate the i^{th} class (positive class) from the $c-1$ others classes (negative class). For very large number of classes, this leads to the extreme unbalance between the positive and the negative class. The problem of binary SVM-SGD comes from **line 4** of Algorithm 1. Let us consider a classification problem with $1,000$ classes, the probability for a positive datapoint sampled is very small (about 0.001) compared with the large chance for a negative datapoint sampled (e.g. 0.999). And then, the binary SVM-SGD classifier focuses mostly on the errors produced by the negative datapoints. Therefore, the binary SVM-SGD classifier has difficulty to separate the positive class from the negative class, well-known as the class imbalance problems.

The survey papers [42–44] present the solutions for dealing with the imbalanced data. At the data level, the algorithms change the class distribution, including over-sampling the minority class [45] or under-sampling the majority class [46,47]. The algorithmic approaches [48–50] are to re-balance the error rate by weighting each type of error with the corresponding cost.

Our balanced batch of binary SVM-SGD (denoted by BBatch-SVM-SGD) belongs to the first approach. For separating the i^{th} class (positive class) from the rest (negative class), the class prior probabilities in this context are highly unequal (e.g. the distribution of the positive class is 0.1% in the $1,000$ classes classification problem), and then over-sampling the minority class is very expensive. We propose the BBatch-SVM-SGD algorithm using under-sampling the majority class (negative class). The training dataset D consists of the positive class D_+ ($|D_+|$ is the cardinality of the positive class) and the negative class D_- ($|D_-|$ is the cardinality of the positive class). Our modification of Algorithm 1 is to use a balanced batch (instead of a datapoint at line 4 of Algorithm 1) to update the w at epoch t. The balanced batch (denoted by BB) includes a datapoint randomly sampling from the positive class D_+ and $k\sqrt{\frac{|D_-|}{|D_+|}}$ datapoints sampling without replacement from the negative class D_-. As illustrated by

[51,52], SGD with a such mini-batch setting can asymptotically achieve optimal speed-up with the average sub-gradients for updating the predictor. Therefore, the updating rule (**lines 6–10** of Algorithm 1) uses the average hinge loss on datapoints in the balanced batch BB and then the classifier is the tail averaged $\bar{w}_t$ on all w_t. The BBatch-SVM-SGD in Algorithm 2 is to separate the i^{th} class (positive class) from the rest (negative class).

Algorithm 2. Training balanced batch of binary SVM-SGD classifier used in the one-versus-all approach of large-scale multi-class SVM

```
input :
        training data of the positive class $D_+$
        training data of the negative class $D_-$
        positive constant $\lambda > 0$
        number of epochs $T$
output:
        hyperplane $w$
begin
    init $w_1 = 0$
    for $t \leftarrow 1$ to $T$ do
        creating a balanced batch $BB_t$ by sampling without replacement $D'_-$ from dataset $D_-$ (with $|D'_-| = k\sqrt{\frac{|D_-|}{|D_+|}}$) and a datapoint from dataset $D_+$
        set $\eta_t = \frac{1}{\lambda t}$
        $BB_k = \{[x_i, y_i] \in BB_t : y_i(w_t.x_i) < 1\}$
        $w_{t+1} = (1 - \eta_t \lambda) w_t + \frac{\eta_t}{|BB_t|} \sum_{[x_i, y_i] \in BB_k} y_i x_i$
    end
    return $\bar{w}_t = \frac{2}{T} \sum_{t=\lfloor \frac{T}{2} \rfloor + 1}^{T} w_t$
end
```

3.2 Parallel Training of BBatch-SVM-SGD

Although BBatch-SVM-SGD classifies very large dataset with high speed, but it does not take the benefits of high performance computing. Furthermore, BBatch-SVM-SGD independently trains c binary classifiers for c classes in multi-class SVM. This is a nice property for parallel learning. The main idea is to learn c binary classifiers in parallel to speedup training tasks of multi-class SVM-SGD. The simplest development of parallel BBatch-SVM-SGD described in Algorithm 3 is based on the shared memory multiprocessing programming model OpenMP on multi-core computers.

The parallel training algorithm of multi-class SVM-SGD (denoted by Par-MC-SVM-SGD) uses Algorithms 2 and 3 for handling large-scale multi-class datasets.

Algorithm 3. Parallel training of BBatch-SVM-SGD in the one-versus-all approach of large-scale multi-class SVM

```
input  : $D$ the training dataset with $c$ classes
output: SVM-SGD model

Learning:
#pragma omp parallel for
for $c_i \leftarrow 1$ to $c$ do                               /* class $c_i$ */
    training BBatch-SVM-SGD($c_i - vs - all$)
end
```

3.3 Incremental Training of Multi-class SVM-SGD in Parallel

The Par-MC-SVM-SGD algorithm needs loading whole dataset in the memory to train the classification models. For very large-scale multi-class datasets such as ImageNet [30] with more than 14 million images and $21,841$ classes, the Par-MC-SVM-SGD algorithm requires at least 350 GB RAM. Any classification algorithm has some difficulties to deal with the challenge of large datasets.

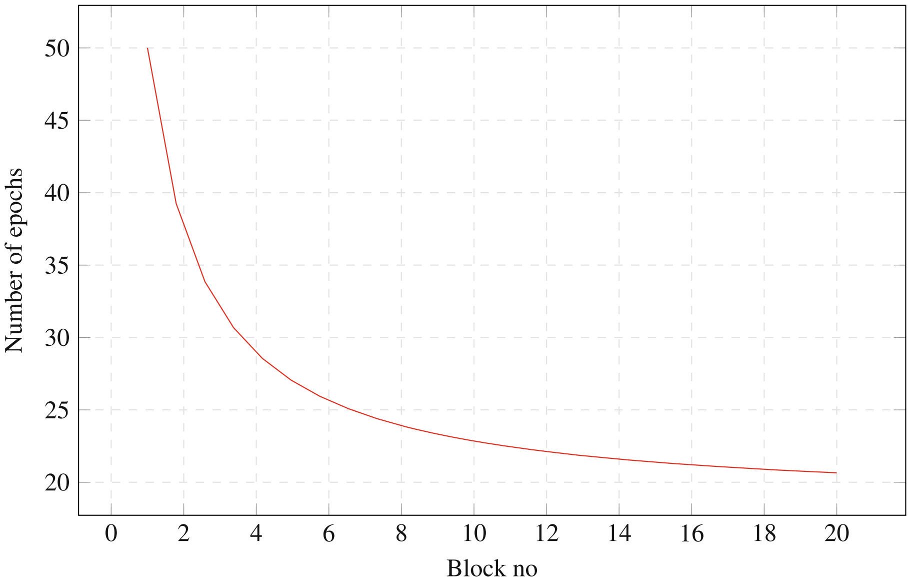


Fig. 4. Number of epochs for MC-SVM-SGD training at block D_b

The incremental training of Par-MC-SVM-SGD (called Incr-Par-MC-SVM-SGD) avoids loading the whole large dataset in main memory: only subsets of the data are considered at any one time and update the solution in growing training set. Let us consider a very large dataset D decomposed into B small blocks of rows, $\{D_1, D_2, \ldots, D_B\}$. For beginning, the Incr-Par-MC-SVM-SGD loads D_1 to learn a multi-class model $mc\text{-}svm\text{-}sgd_1$ with the Par-MC-SVM-SGD. At step b, the Incr-Par-MC-SVM-SGD uses D_b and datapoints in D_{b-1} near from separating boundary (ref. to $mc\text{-}svm\text{-}sgd_{b-1}$) to train a multi-class model $mc\text{-}svm\text{-}sgd_b$ with the Par-MC-SVM-SGD. We remark that the Incr-Par-MC-SVM-SGD aims at updating the previous model $mc\text{-}svm\text{-}sgd_{b-1}$ in growing training set. Hence, the Incr-Par-MC-SVM-SGD trains the model at the next step with the number of epochs getting decreased (for example in Fig. 4). The last model $mc\text{-}svm\text{-}sgd_B$ is the final multi-class classifier.

Algorithm 4. Incremental training of Par-MC-SVM-SGD

```
input :
        training data $D = \{D_1, D_2, \ldots, D_B\}$
        positive constant $\lambda > 0$
        number of epochs $T$
output:
        MC-SVM-SGD model
begin
    init $mc\text{-}svm\text{-}sgd_1$ = Par-MC-SVM-SGD($D_1, \lambda, T$)
    for $b \leftarrow 2$ to $B$ do
        training sample $S_b$ includes $D_b$ and datapoints in $D_{b-1}$ near from
        separating boundary (ref. to $mc\text{-}svm\text{-}sgd_{b-1}$)
        $mc\text{-}svm\text{-}sgd_b$ = Par-MC-SVM-SGD($S_b, \lambda, T[\frac{b+1}{b+2}]^{b-1}$)
    end
    return $mc\text{-}svm\text{-}sgd_B$
end
```

4 Evaluation

In order to evaluate the performance of the new incremental parallel multi-class SVM-SGD (Incr-Par-MC-SVM-SGD) algorithm for classifying large amounts of images into many classes, we have implemented the Incr-Par-MC-SVM-SGD and the Par-MC-SVM-SGD (batch version loading whole dataset in the memory) in C/C++ using the SGD library [24]. We are interested in two recent algorithms, LIBLINEAR (a library for large linear classification [40], the parallel version on multi-core computers) and OCAS (an optimized cutting plane algorithm for SVM [53]) because they are well-known as highly efficient standard linear SVM. Our comparison is reported in terms of correctness, training time and memory requirements obtained by Incr-Par-MC-SVM-SGD, Par-MC-SVM-SGD, LIBLINEAR and OCAS.

All experiments are run on machine Linux Fedora 20, Intel(R) Core i7-4790 CPU, 3.6 GHz, 4 cores and 32 GB main memory.

4.1 Datasets

The Incr-Par-MC-SVM-SGD algorithm is designed for the large number of images with many classes, so we have evaluated its performance on the three following datasets.

ImageNet 10. This dataset contains the 10 largest classes from ImageNet [30], including 24,807 images with size 2.4 GB. In each class, we sample 90 % images for training and 10 % images for testing (with random guess 10 %). First, we construct BoW of every image using dense SIFT descriptor (extracting SIFT on a dense grid of locations at a fixed scale and orientation) and 5,000 codewords. Then, we use feature mapping from [54] to get the high-dimensional image representation in 15,000 dimensions. This feature mapping has been proven to give a good image classification performance with linear classifiers [54]. We end up with 2.6 GB of training data.

ImageNet 100. This dataset consists of the 100 largest classes from ImageNet [30], including 183,116 images with size 23.6 GB. In each class, we sample 50 % images for training and 50 % images for testing (with random guess 1 %). We also construct BoW of every image using dense SIFT descriptor and 5,000 codewords. For feature mapping, we use the same method as we do with ImageNet 10. The final size of training data is 8 GB.

ILSVRC 2010. This dataset contains 1,000 classes from ImageNet [30], including 1.2 million images ($\sim$ 126 GB) for training, 50 thousand images ($\sim$ 5.3 GB) for validation and 150 thousand images ($\sim$ 16 GB) for testing. We use BoW feature set provided by [30] and the method reported in [55] to encode every image as a vector in 21,000 dimensions. We take roughly 900 images per class for training dataset, so the total training images is 887,816 and the training data size is about 12.5 GB. All testing samples are used to test SVM models. Note that the random guess performance of this dataset is 0.1 %.

4.2 Parameters

The positive constant $C = 1,000,000$ (a trade-off between the margin size and the errors in learning SVM algorithms, the same tuning in [12,26,27]) was used in LIBLINEAR and OCAS to train classification models.

Our Incr-Par-MC-SVM-SGD and Par-MC-SVM-SGD algorithms learn the balanced batch stochastic gradient descend of SVM (BBatch-SVM-SGD) with $T = 50$ epochs and regularization term $\lambda = 0.00002$. Furthermore, the Incr-Par-MC-SVM-SGD loads in the main memory the small blocks of rows (instead of the whole dataset) to learn the classification model in incremental way. The large-scale multi-class dataset should be split into the small enough blocks of rows ($\sim$ 10 % – 20 % of the full datasets for a compromise the classification accuracy and the main memory usage). And then, the block sizes of ImageNet 10, ImageNet 100 and ILSVRC 2010 datasets are set to $2,000$, $15,000$ and $200,000$, respectively.

Due to the PC (Intel(R) Core i7-4790 CPU, 4 cores) used in the experimental setup, we try to vary the number of OpenMP threads (1, 4, 8 threads) for all training tasks.

4.3 Classificaton Results

Firstly, we are interested in the performance comparison in terms of training time, memory usage and accuracy.

Memory Usage. The main memory usage of training algorithms is presented in Table 1 and Fig. 5. As it was expected, the Incr-Par-MC-SVM-SGD uses less memory than other algorithms.

Regarding the comparison of the Incr-Par-MC-SVM-SGD with OCAS, one can see that the gains of main memory requirements ensured by the Incr-Par-MC-SVM-SGD against OCAS are 86.66 %, 81.65 % and 90.04 % for ImageNet 10, ImageNet 100 and ILSVRC 2010, respectively.

The improvements of main memory used by the Incr-Par-MC-SVM-SGD against LIBLINEAR correspond to 89.14 %, 87.05 % and 68.72 % for ImageNet 10, ImageNet 100 and ILSVRC 2010.

In the comparison with the Par-MC-SVM-SGD (batch version loading whole dataset in the memory), the Incr-Par-MC-SVM-SGD saves up 89.14 %, 84.80 % and 69.71 % main memory for ImageNet 10, ImageNet 100 and ILSVRC 2010, respectively.

Table 1. Memory usage (GB) of training algorithms

Dataset	ImageNet 10	ImageNet 100	ILSVRC 2010
OCAS	2.55	7.90	52.90
LIBLINEAR	3.13	11.20	16.90
Par-MC-SVM-SGD	3.13	9.54	17.40
Incr-Par-MC-SVM-SGD	0.34	1.45	5.27

Training Time. Table 2 and Fig. 6 present the training time of algorithms for ImageNet 10 (the small multi-class dataset). The Incr-Par-MC-SVM-SGD with 8 OpenMP threads is 82.05 times faster than OCAS (running on 1 core) and slight faster than LIBLINEAR (with 8 OpenMP threads). As mentioned in Algorithm 4, the Incr-Par-MC-SVM-SGD needs learning the datapoints near from separating boundary more than the Par-MC-SVM-SGD. The Par-MC-SVM-SGD performs 2 times faster than the Incr-Par-MC-SVM-SGD.

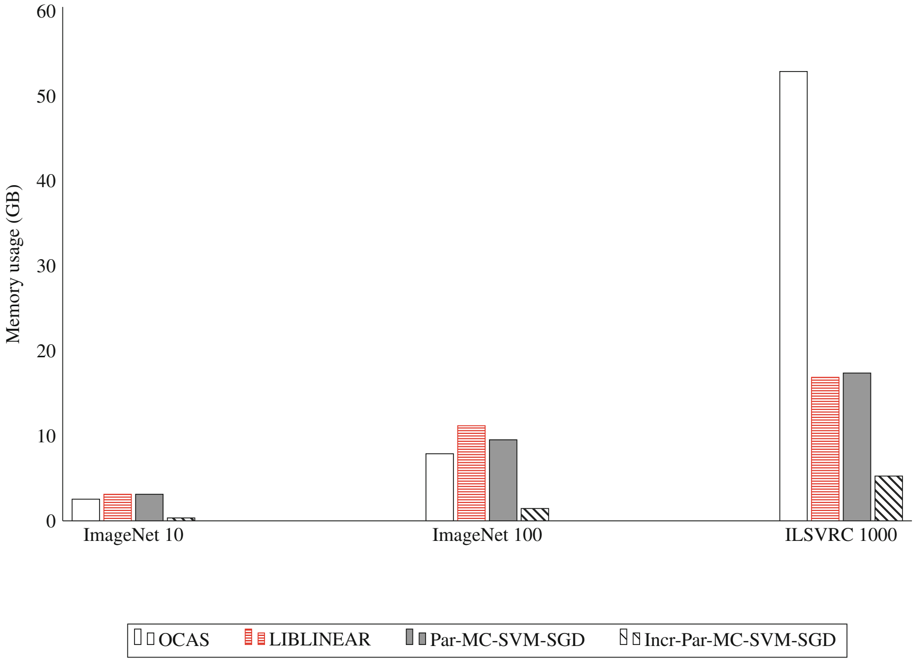


Fig. 5. Memory usage (GB) of training algorithms

The training time of algorithms on ImageNet 100 presented in Table 3 and Fig. 8 show that the Incr-Par-MC-SVM-SGD achieves a significant speed-up in learning process using 8 OpenMP threads. It is 74.62 times faster than OCAS and 2.22 times faster than LIBLINEAR. Once again, the Par-MC-SVM-SGD is 2 times faster than the Incr-Par-MC-SVM-SGD.

ILSVRC 2010 has large amount of images (more than 1 million images) and very large number of classes (1,000 classes). Therefore, OCAS has not finished the learning task in several days. LIBLINEAR with 8 OpenMP threads takes 1,004.00 min to train the classification model for this dataset. Our Incr-Par-MC-SVM-SGD algorithm performs the learning task in 50.35 min with 8

Table 2. Training time (minutes) on ImageNet 10

Algorithm	# OpenMP threads		
	1	4	8
OCAS	106.67		
LIBLINEAR	3.12	1.50	1.48
Par-MC-SVM-SGD	1.85	0.67	0.66
Incr-Par-MC-SVM-SGD	3.86	1.37	1.30

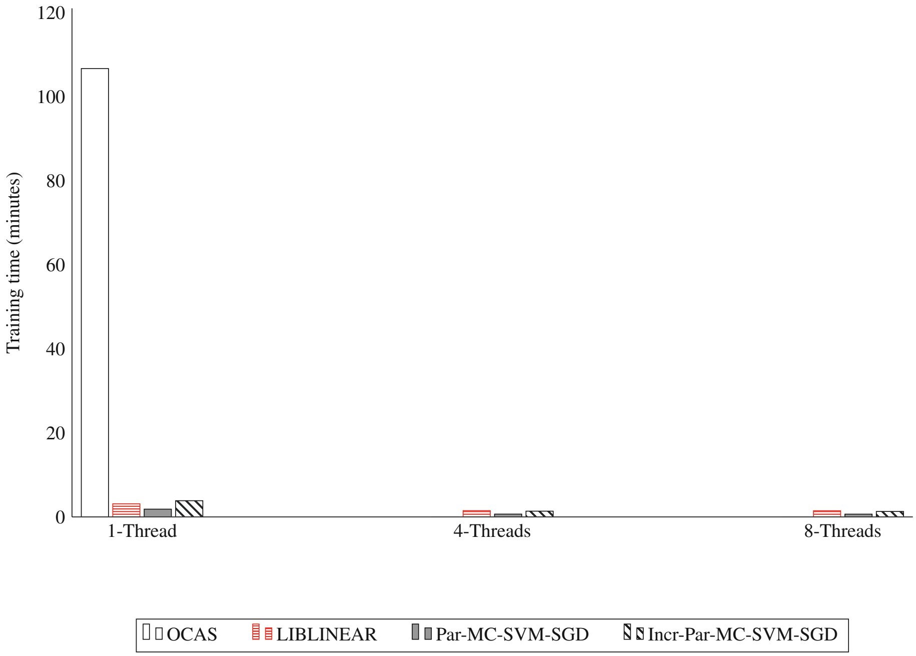


Fig. 6. Training time (minutes) on ImageNet 10

Table 3. Training time (minutes) on ImageNet 100

Algorithm	# OpenMP threads		
	1	4	8
OCAS	1016.35		
LIBLINEAR	63.42	30.49	30.18
Par-MC-SVM-SGD	24.66	7.03	6.91
Incr-Par-MC-SVM-SGD	47.09	14.86	13.62

OpenMP threads. This indicates that the Incr-Par-MC-SVM-SGD is 19.94 times faster than LIBLINEAR. The Incr-Par-MC-SVM-SGD needs 5.5 min more than the Par-MC-SVM-SGD for the learning task (Fig. 7).

Due to Intel(R) i7-4790 processor (4 cores), almost parallel algorithms using 8 threads can not improve much the training time against the 4 threads setting (Table 4).

Classification accuracy. The classification results in terms of accuracy presented in Table 5 and Fig. 9 show that the training of the Incr-Par-MC-SVM-SGD in incremental way has very few compromise the correctness, compared to its batch training of the Par-MC-SVM-SGD.

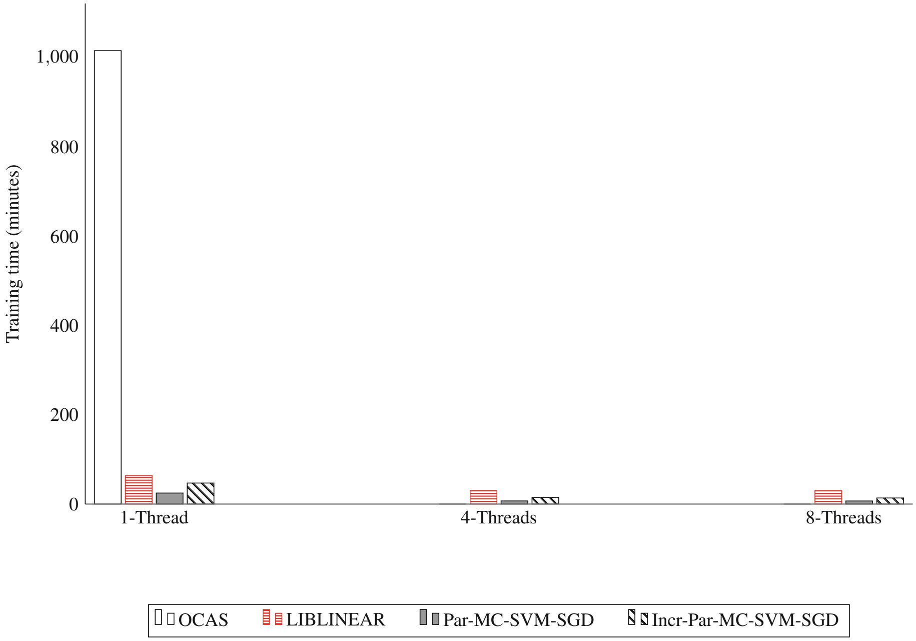


Fig. 7. Training time (minutes) on ImageNet 100

Table 4. Training time (minutes) on ILSVRC 2010

Algorithm	# OpenMP threads		
	1	4	8
OCAS	N/A		
LIBLINEAR	3106.48	1037.00	1004.00
Par-MC-SVM-SGD	188.70	51.79	44.85
Incr-Par-MC-SVM-SGD	206.68	61.98	50.35

The Incr-Par-MC-SVM-SGD is more accurate than OCAS on ImageNet 10 while making more classification mistakes than OCAS on ImageNet 100. The Incr-Par-MC-SVM-SGD also achieves very competitive performances compared to Par-MC-SVM-SGD and LIBLINEAR on ImageNet 10 and ImageNet 100.

ILSVRC 2010 is a large dataset (with more than 1 million images and 1,000 classes). Thus, it is very difficult for many state-of-the-art algorithms to obtain a high rate in classification performance. In particular, with the feature set provided by ILSVRC 2010 competition the state-of-the-art system [30,56] reports an accuracy of approximately 19 % (it is far above random guess, 0.1 %).

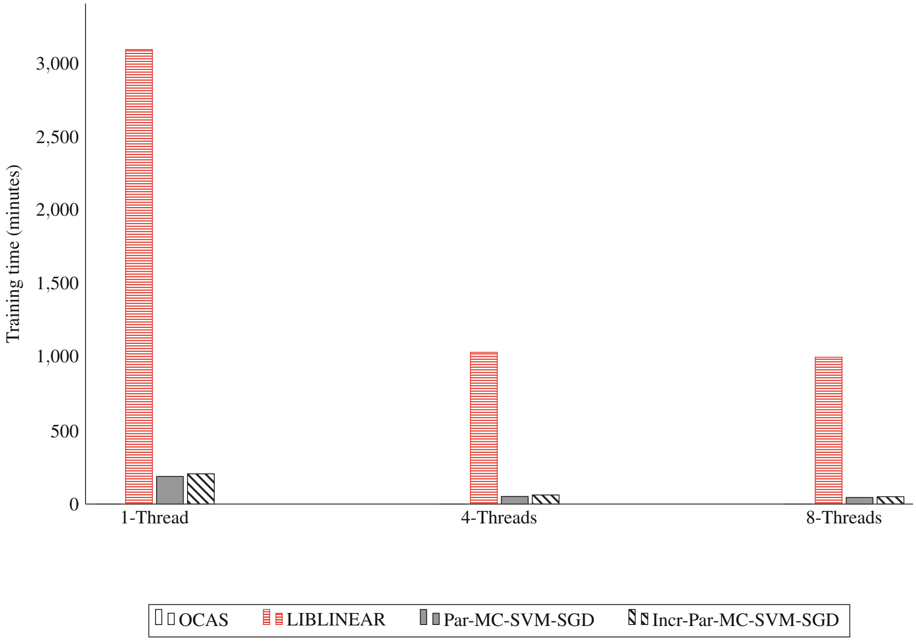


Fig. 8. Training time (minutes) on ILSVRC 2010

Our Incr-Par-MC-SVM-SGD algorithm gives a higher accuracy rate than [30,56] with the same feature set (21.19 % vs. 19 %). The Incr-Par-MC-SVM-SGD holds the rank 2 after the Par-MC-SVM-SGD. Note that the Incr-Par-MC-SVM-SGD learns much faster than LIBLINEAR while maintaining a high correctness rate. These results show that our Incr-Par-MC-SVM-SGD has a great ability to scale-up to full ImageNet dataset.

Table 5. Overall classification accuracy (%)

Dataset	ImageNet 10	ImageNet 100	ILSVRC 2010
OCAS	72.07	52.75	N/A
LIBLINEAR	75.09	54.07	21.11
Par-MC-SVM-SGD	75.33	53.60	21.90
Incr-Par-MC-SVM-SGD	74.16	51.98	21.19

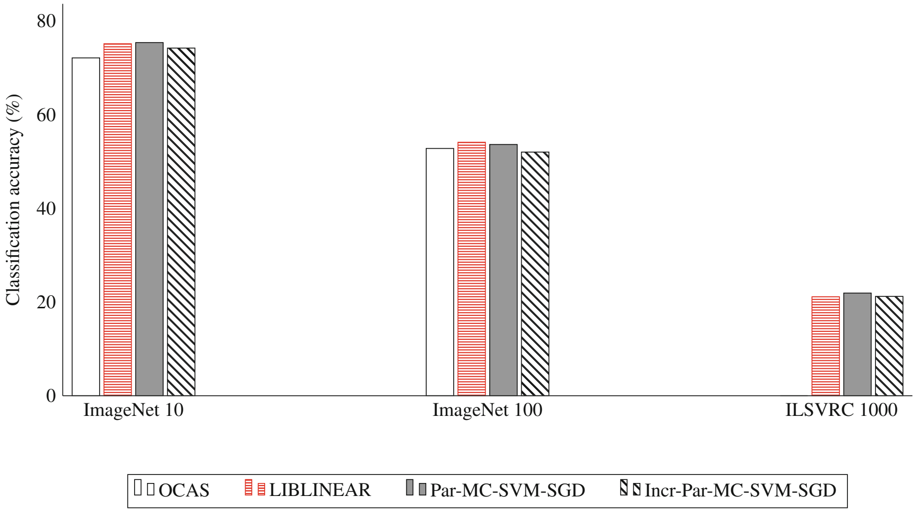


Fig. 9. Overall classification accuracy (%)

5 Conclusion and Future Works

We have presented the new incremental parallel multi-class SVM-SGD that achieves high performances for dealing with large amounts of images and large-scale multi-class on PCs. The balanced batch SGD of SVM (BBatch-SVM-SGD) is proposed for trainning two-class classifiers used in the multi-class problems. The incremental training process of classifiers in parallel way on multi-core computers is also developped for efficiently classifying large image datasets into very large number of classes. Our algorithm is evaluated on the 10, 100 and 1,000 largest classes of ImageNet datasets. The incremental algorithm saves up from 68.72 % to 90.04 % main memory usage while achieving significant low cost in terms of training time without (or very few) compromise the classification accuracy. It is able to handle in high-speed training the dataset larger than the memory capacity of PCs.

In the future, we will provide more empirical test on full ImageNet dataset with $21,000$ classes. We also intend to develop distributed MC-SVM-SGD algorithms for efficiently dealing with large scale multi-class problems on Spark [57].

References

1. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: 9th IEEE International Conference on Computer Vision (ICCV 2003), 14–17, October 2003, Nice, France, pp. 1470–1477 (2003)
2. Li, F., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20–26 June 2005, San Diego, CA, USA, pp. 524–531 (2005)

3. Lowe, D.G.: Object recognition from local scale invariant features. In: Proceedings of the 7th International Conference on Computer Vision, pp. 1150–1157 (1999)
4. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
5. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, vol. 1, pp. 281–297. University of California Press, January 1967
6. Vapnik, V.: The Nature of Statistical Learning Theory. Springer-Verlag, New York (1995)
7. Platt, J.: Fast training of support vector machines using sequential minimal optimization. In: Schölkopf, B., Burges, C., Smola, A. (eds.) Advances in Kernel Methods Support Vector Learning, pp. 185–208 (1999)
8. Boser, B., Guyon, I., Vapnik, V.: An training algorithm for optimal margin classifiers. In: Proceedings of 5th ACM Annual Workshop on Computational Learning Theory of 5th ACM Annual Workshop on Computational Learning Theory, pp. 144–152. ACM (1992)
9. Syed, N., Liu, H., Sung, K.: Incremental learning with support vector machines. In: Proceedings of the ACM SIGKDD International Conference on KDD. ACM (1999)
10. Do, T.N., Poulet, F.: Incremental SVM and visualization tools for bio-medical data mining. In: Proceedings of the European Workshop on Data Mining and Text Mining for Bioinformatics, pp. 14–19 (2003)
11. Yu, H., Hsieh, C., Chang, K., Lin, C.: Large linear classification when data cannot fit in memory. ACM Trans. Knowl. Discov. Data **5**(4), 23: 1–23: 23 (2012)
12. Doan, T.N., Do, T.N., Poulet, F.: Large scale classifiers for visual classification tasks. Multimedia Tools Appl. **74**(4), 1199–1224 (2015)
13. Poulet, F., Do, T.N.: Mining very large datasets with support vector machine algorithms. In: Camp, O., Filipe, J., Hammoudi, S., Piattini, M. (eds.) Enterprise Information Systems V, pp. 177–184 (2004)
14. Do, T.N., Poulet, F.: Classifying one billion data with a new distributed svm algorithm. In: RIVF, pp. 59–66 (2006)
15. Do, T.N., Nguyen, V.H.: A novel speed-up svm algorithm for massive classification tasks. In: IEEE International Conference on Research, Innovation and Vision for the Future, RIVF 2008, pp. 215–220. IEEE (2008)
16. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. In: proceedings of the 17th International Conference on Machine Learning, pp. 999–1006. ACM (2000)
17. Do, T.N., Poulet, F.: Mining very large datasets with SVM and visualization. In: proceedings of 7th International Conference on Entreprise Information Systems, pp. 127–134 (2005)
18. Bordes, A., Ertekin, S., Weston, J., Bottou, L.: Fast kernel classifiers with online and active learning. J. Mach. Learn. Res. **6**, 1579–1619 (2005)
19. Do, T.N., Le Thi, H.A.: Massive classification with support vector machines. In: Nguyen, N.T. (ed.) Transactions on Computational Collective Intelligence XVIII. LNCS, vol. 9240, pp. 147–165. Springer, Heidelberg (2015). doi:10.1007/978-3-662-48145-5_8
20. Segata, N., Blanzieri, E.: Fast local support vector machines for large datasets. In: Perner, P. (ed.) MLDM 2009. LNCS (LNAI), vol. 5632, pp. 295–310. Springer, Heidelberg (2009). doi:10.1007/978-3-642-03070-3_22

21. Do, T.-N.: Non-linear classification of massive datasets with a parallel algorithm of local support vector machines. In: Le Thi, H.A., Nguyen, N.T., Do, T.V. (eds.) Advanced Computational Methods for Knowledge Engineering. AISC, vol. 358, pp. 231–241. Springer, Heidelberg (2015)
22. Do, T.-N., Poulet, F.: Random local SVMs for classifying large datasets. In: Dang, T.K., Wagner, R., Küng, J., Thoai, N., Takizawa, M., Neuhold, E. (eds.) FDSE 2015. LNCS, vol. 9446, pp. 3–15. Springer, Heidelberg (2015). doi:10.1007/978-3-319-26135-5_1
23. Shalev-Shwartz, S., Singer, Y., Srebro, N.: Pegasos: primal estimated sub-gradient solver for SVM. In: Proceedings of the Twenty-Fourth International Conference Machine Learning, pp. 807–814. ACM (2007)
24. Bottou, L., Bousquet, O.: The tradeoffs of large scale learning. In: Platt, J., Koller, D., Singer, Y., Roweis, S. (eds.) Advances in Neural Information Processing Systems, vol. 20, pp. 161–168 (2008)
25. Sánchez, J., Perronnin, F.: High-dimensional signature compression for large-scale image classification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1665–1672 (2011)
26. Do, T.N.: Parallel multiclass stochastic gradient descent algorithms for classifying million images with very-high-dimensional signatures into thousands classes. Vietnam J. Comput. Sci. **1**(2), 107–115 (2014)
27. Do, T.-N., Poulet, F.: Parallel multiclass logistic regression for classifying large scale image datasets. In: Le Thi, H.A., Nguyen, N.T., Do, T.V. (eds.) Advanced Computational Methods for Knowledge Engineering. AISC, vol. 358, pp. 255–266. Springer, Heidelberg (2015)
28. Li, F.F., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. Comput. Vis. Image Underst. **106**(1), 59–70 (2007)
29. Griffin, G., Holub, A., Perona, P.: Caltech-256 Object Category Dataset. Technical Report CNS-TR-2007-001. California Institute of Technology (2007)
30. Deng, J., Berg, A.C., Li, K., Li, F.F.: What does classifying more than 10, 000 image categories tell us? In: European Conference on Computer Vision, pp. 71–84 (2010)
31. Doan, T.-N., Do, T.-N., Poulet, F.: Large scale image classification with many classes, multi-features and very high-dimensional signatures. In: Nguyen, N.T., van Do, T., Thi, H.A. (eds.) ICCSAMA 2013. SCI, vol. 479, pp. 105–116. Springer, Heidelberg (2013)
32. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, New York (2000)
33. Ben-Akiva, M., Lerman, S.: Discrete Choice Analysis: Theory and Application to Travel Demand. The MIT Press, Cambridge (1985)
34. Weston, J., Watkins, C.: Support vector machines for multi-class pattern recognition. In: Proceedings of the Seventh European Symposium on Artificial Neural Networks, pp. 219–224 (1999)
35. Guermeur, Y.: VC theory of large margin multi-category classifiers. J. Mach. Learn. Res. **8**, 2551–2594 (2007)
36. Kreßel, U.: Pairwise classification and support vector machines, Advances in Kernel Methods: Support Vector Learning, pp. 255–268 (1999)
37. Vural, V., Dy, J.: A hierarchical method for multi-class support vector machines. In: Proceedings of the Twenty-First International Conference on Machine Learning, pp. 831–838 (2004)

38. Benabdeslem, K., Bennani, Y.: Dendogram-based svm for multi-class classification. J. Comput. Inf. Technol. **14**(4), 283–289 (2006)
39. Do, T.N., Lenca, P., Lallich, S.: Classifying many-class high-dimensional fingerprint datasets using random forest of oblique decision trees. Vietnam J. Comput. Sci. **2**(1), 3–12 (2015)
40. Fan, R., Chang, K., Hsieh, C., Wang, X., Lin, C.: LIBLINEAR: a library for large linear classification. J. Mach. Learn. Res. **9**(4), 1871–1874 (2008)
41. Chang, C.C., Lin, C.J.: LIBSVM : a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(27), 1–27 (2011)
42. Japkowicz, N. (ed.): AAAI'Workshop on Learning from Imbalanced Data Sets. Number WS-00-05 in AAAI Tech Report (2000)
43. Weiss, G.M., Provost, F.: Learning when training data are costly: the effect of class distribution on tree induction. J. Artif. Intell. Res. **19**, 315–354 (2003)
44. Visa, S., Ralescu, A.: Issues in mining imbalanced data sets - a review paper. In: Midwest Artificial Intelligence and Cognitive Science Conference, Dayton, USA, pp. 67–73 (2005)
45. Chawla, N.V., Lazarevic, A., Hall, L.O., Bowyer, K.W.: SMOTEBoost: improving prediction of the minority class in boosting. In: Lavrač, N., Gamberger, D., Todorovski, L., Blockeel, H. (eds.) PKDD 2003. LNCS (LNAI), vol. 2838, pp. 107–119. Springer, Heidelberg (2003)
46. Liu, X.Y., Wu, J., Zhou, Z.H.: Exploratory undersampling for class-imbalance learning. IEEE Trans. Syst. Man Cybern. Part B **39**(2), 539–550 (2009)
47. Ricamato, M.T., Marrocco, C., Tortorella, F.: Mcs-based balancing techniques for skewed classes: an empirical comparison. In: ICPR, pp. 1–4 (2008)
48. Domingos, P.: Metacost: a general method for making classifiers cost sensitive. In: International Conference on Knowledge Discovery and Data Mining, pp. 155–164 (1999)
49. Zhou, Z.H., Liu, X.Y.: On multi-class cost-sensitive learning. In: 21st National Conference on Artificial Intelligence, Boston, MA, USA, pp. 567–572 (2006)
50. Wang, B.X., Japkowicz, N.: Boosting support vector machines for imbalanced data sets. Knowl. Inf. Syst. **25**(1), 1–20 (2010)
51. Cotter, A., Shamir, O., Srebro, N., Sridharan, K.: Better mini-batch algorithms via accelerated gradient methods. In: Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, pp. 1647–1655 (2011)
52. Li, M., Zhang, T., Chen, Y., Smola, A.J.: Efficient mini-batch training for stochastic optimization. In: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2014, pp. 661–670 (2014)
53. Franc, V., Sonnenburg, S.: Optimized cutting plane algorithm for large-scale risk minimization. J. Mach. Learn. Res. **10**, 2157–2192 (2009)
54. Vedaldi, A., Zisserman, A.: Efficient additive kernels via explicit feature maps. IEEE Trans. Pattern Anal. Mach. Intell. **34**(3), 480–492 (2012)
55. Wu, J.: Power mean svm for large scale visual classification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2344–2351 (2012)
56. Berg, A., Deng, J., Li, F.F.: Large scale visual recognition challenge 2010, Technical report (2010)
57. Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., Stoica, I.: Spark: cluster computing with working sets. In: Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, USENIX Association (2010)

A Machine Learning-Based Approach for Predicting the Execution Time of CFD Applications on Cloud Computing Environment

Duong Ngoc Hieu[1], Thai Tieu Minh[2], Trinh Van Quang[1],
Bui Xuan Giang[1], and Tran Van Hoai[1](✉)

[1] Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
hoai@hcmut.edu.vn

[2] Vietnam Academy of Science and Technology, Ho Chi Minh City, Vietnam

Abstract. Calibrations and validations of Computational Fluid Dynamics (CFD) applications are significantly time-consuming. To reduce the execution time of the CFD applications, parallel-computing approach is often employed. In addition, high performance computing systems and cloud computing solutions are also appropriate tools to the CFD applications. One of the challenging problems is to schedule tasks on virtualized machines of the cloud-based high performance systems. Instead of employing an adaptive algorithm to cope with the uncertainty of the virtualized resources, in this study, we propose an idea to predict the execution time of Telemac-2D, which is a CFD application. The predicted execution time is very essential in all scheduling algorithms. The application is executed several times with different settings of model's parameters and allocated resources to produce an experimental dataset. The dataset is then used to predict the execution time of the application by utilizing a machine learning-based approach. The predictive model consists of two steps that classify and predict the execution. The C4.5 algorithm is used to classify the execution ending status whereas Multi-layer Perceptron (MLP) and a mixture of MLPs (MiMLP) are used to predict the execution time. The experiments indicate that the predictive model is appropriate to predict the execution of the Telemac-2D application since the accuracy of the C4.5 algorithm is 100 % and R and MARE of MiMLP are 0.957 and 17.090, respectively.

Keywords: Multi-layer perceptron · Cloud computing · Computational fluid dynamics · Telemac

1 Introduction

Recently, cloud computing has been commonly used as an effective solution to deal with several problems involving high computational complexity. For problems requiring high performance computing, especially Computational Fluid Dynamics (CFD), ones often deploy their solutions on cloud computing services,

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 40–52, 2016.
DOI: 10.1007/978-3-319-48057-2_3

such as CFD Direct From the Cloud, Caedium RANS Flow, ANSYS Enterprise Cloud, and SimuCloud. With these cloud computing services, the CFD applications are deployed quickly, easily, and without considering any setting adjustments and allocated resources.

However, primary challenges of the CFD applications are time-consuming during calibrations and validations. The calibration is a process to adjust parameters of models until the agreement between model's results and experimental data is reached. Then, in the validation step, the calibrated model is verified with another dataset [11,15]. To reduce the time of calibrations and validations, parallel-computing approach is often considered. However, the speed-up time is not proportional with computing units that are used in parallel algorithms [8].

Optimization of resource provisioning in cloud computing is a difficult problem by its own undetermined nature [1,2]. Doyle *et al.* indicated that there were three steps to effectively manage cloud resources including predetermined time-to-completion of workloads, accurate resource prediction (ARP), and effective control of the number of cloud instances servicing workloads [3]. The authors also concluded that it is hard to predict exactly time-to-completion because there are many factors impacting the execution time. These factors consist of the specifics of workloads, computing-unit reservation mechanisms, networking-transport layers, and so on.

It is agreed that time-to-completion is hard to be predicted, but some approximation of this parameter will be very useful for a scheduling algorithm (or a high performance computing system) which is specialized for domain-specific applications. In this study, we attempt to predict the execution time of a specific CFD application called Telemac-2D application. The Telemac-2D application is used to simulate flood diversion in the MeKong Delta. The application is executed several times with different settings of model's parameters and allocated resources to produce an experimental dataset. The dataset is then used to predict the execution time of the application by utilizing a machine learning-based approach. The predictive model consists of two steps that classify and predict the execution. The C4.5 algorithm is used to classify the execution ending status whereas Multi-layer Perceptron (MLP) and a mixture of MLPs (MiMLP) are used to predict the execution time.

The rest of this paper is organized as follows. Section 2 presents some related work. In Sect. 3, we introduce the context of this study including a cloud computing environment, the Telemac-2D application, the study area of the application, and an experimental dataset collected from the application. Section 4 presents a predictive model based on decision algorithms and artificial neural networks. The experimental results are reported in Sect. 5. Finally, we draw conclusion in Sect. 6.

2 Related Work

In the past few years, there has been a great deal of research that applies machine learning methods for predicting the execution time of computer programs. Through analyzing the source code of these programs and employed facility, Huang *et al.* proposed two Sparse POlynomial REgression algorithms that discover relationships between the execution time and features extracted from the source code without any expertise [6]. To assess the performance of the two algorithms, they used three case studies including Lucene Search Engine, Find Maxima, and Segmentation in ImageJ framework.

In a heterogeneous computing environment, Priya *et al.* [12] attempted to predict the execution time of machine learning algorithms based on a meta-learning approach. Then, according to the predicted execution time, the authors utilized Genetic Algorithm to schedule computational tasks. A significant achievement of their work is to discover relationships between independent parameters and the execution time.

In addition to the machine learning approach, Matsunaga and Fortes built a novel method called PQR2 for predicting execution time. The authors deployed the method for two bioinformatic applications. Through experiments, they indicated that PQR2 outperforms Support Vector Machines and K-Nearest Neighbours [10]. Ipek *et al.* [7] focused on predicting the performance of parallel applications, namely SMG2000. In their study, a neural network-based model which is trained by a back-propagation algorithm is utilized to tackle the issue. Besides, Kasperkiewicz *et al.* combined neural networks and fuzzy methods for predicting strength properties of high-performance concrete mixes [9].

3 Context

3.1 PaaS Scheduling Platform

We build a cloud-based high performance computing system, namely PaaS Scheduling Platform (PSSP), based on OpenStack platform in 4 HP computing nodes. PSSP consists of one controller node and three computing nodes as seen in Fig. 1. In the controller node, a sharing file system is implemented to create a unified storing disk according to distributed physical disks. Given the unified stored disk, CFD applications and MPI libraries are installed in many virtual machines that will take part in the same parallel CFD tasks. The three computing nodes are responsible for deploying virtual machines. When a new virtual machine is created, it communicates with the others via an internal network or Internet.

3.2 OpenTelemac

OpenTelemac[1] is a suite of CFD models to simulate offshore, coastal, rivers, and estuaries. It has been developed by the Artelia collaboration and organizations of Germany and United Kingdom. The main objective of the Telemac suite is to predict phenomena involving flow, wave propagation and sediment transport. Two principal modules of the suite are Telemac-2D (Saint-Venant equations) and Telemac-3D (Navier-Stokes equations). These two modules are combined with

[1] http://www.opentelemac.org/.

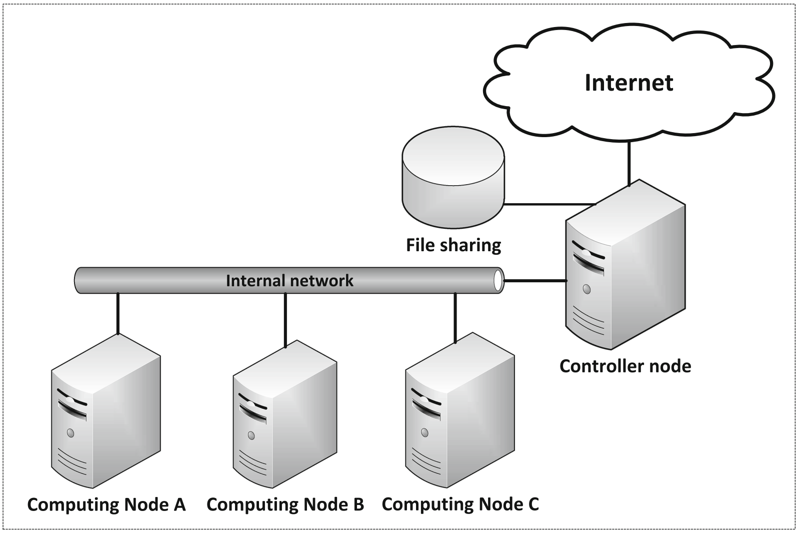


Fig. 1. The physical schema of PSSP

other modules, such as Tomawac, Sisyphe, Artemis, and Dredgesim to model complex hydrologic processes.

In order to run the Telemac-2D application in PSSP, it is installed and configured with the OpenMPI library. When the installation is completed, a snapshot of disk image is stored in PSSP Image Services. The application is able to access necessary data from the unified storing disk. While the application executes, PSSP observes allocated resources used by the application, and stores the information into log files.

Generally, to simulate flood diversion, many methods of the Telemac-2D application are used, such as Conjugate Gradient method on the Normal Equations (CGNE), Minimum Error method (ME), Square Conjugate Gradient method (SCG), Conjugated Residue method (CR), and Generalised Minimum RESidual (GMRES). The execution time of the application dependently varies in the input parameters of the flood diversion, such as domains, time-steps, runtime, convergence conditions, stability conditions, and so on.

3.3 A Case Study

In this study, the flood diversion of the Mekong Delta in Thong *at el.* works [16] has been selected as a case study. This problem is addressed in 2D-numerical modeling to estimate the variation of stage and discharge for some specific scenarios. The results of the simulation are used to assess the impact of the flood diversion in the MeKong Delta. To discretize the modeled area, Finite Element Mesh (FEM) consists of 716.000 elements corresponding to the area of 80850 km^2 (Fig. 2). Note that the temporal resolution of the mesh is 20 s.

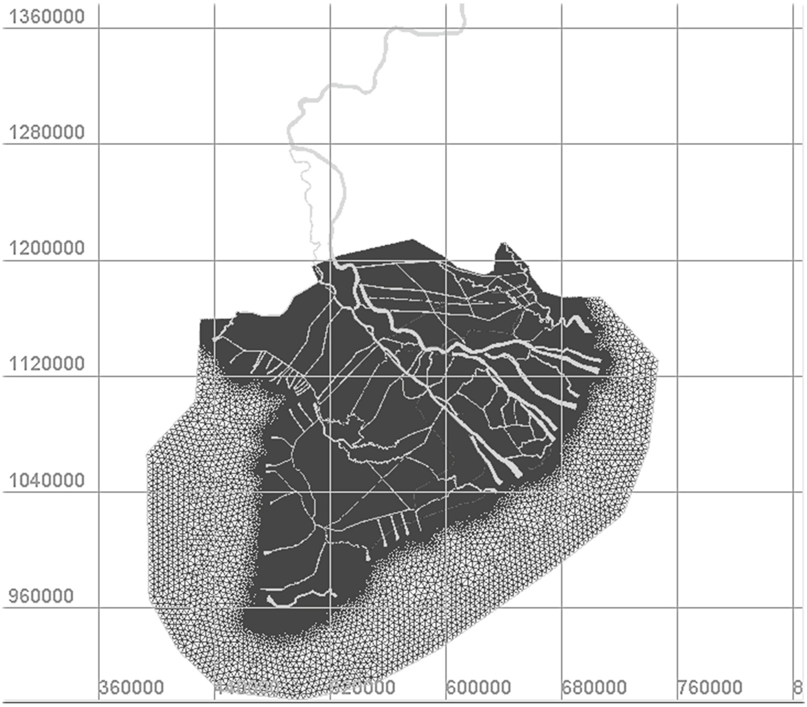


Fig. 2. The Finite Element Mesh with 716.000 elements corresponding to the Mekong Delta [16]

3.4 Experimental Dataset

There is a large number of parameters needed to configure for the Saint-Venant equations in the Telemac-2D application. Typically, the parameters are geometry, friction coefficients, open boundary conditions (tide, discharge, elevation, etc.), initial conditions (water level, current, etc.), tidal harmonic constants, velocity diffusivity, Coriolis coefficients, Chezy number or Manning numbers, and so on. Moreover, depending on different objectives of simulation, several different parameters may be used.

However, principle parameters that impact to the execution time of the Telemac-2D application are the parameters of spatial resolution and temporal resolution. Depending on numerical methods, the time execution may be escalate with the number of FEM elements. The number of time-steps significantly affects the complexity and the execution time. In addition to these parameters, employed methods to solve linear equations in the Telemac-2D application also impact to the execution time.

After analyzing the log files from CFD running processes, several significant parameters are selected as input variables to predict the execution time. These parameters are RAM, the number of processors, the number of virtual machines

Table 1. The sample dataset for training in C4.5 and artificial neutron networks

RAM	Solver name	Parameter 1	Parameter 2	Virtual machines	Processors	Execution time (s)	Status
1024	CGNE	107	51	1	1	221	Yes
4096	ME	95	45	2	16	103	Yes
2048	CGNE	107	51	1	16	5	No
8192	SCG	25	7	4	8	22	Yes
4096	CR	85	37	4	4	62	Yes
1024	GMRES	23	8	2	16	4	No
8192	GMRES	23	8	1	2	27	Yes
2048	ME	95	45	4	8	58	Yes
1024	CGNE	107	51	1	2	101	Yes
8192	CR	85	37	4	8	67	Yes
...	...	...	...	...	...	...	...

(VMs), and solver methods. Table 1 presents the samples of the experimental dataset. In addition to the solver methods, we discretize them by their statistical characteristics, namely Parameter 1 and Parameter 2. The Parameter 1 of the solver method is derived from the average value of the execution time whereas the Parameter 2 is derived from their standard deviation.

4 Methodology

According to the characteristics of the collected dataset including two types of outputs: execution ending status and execution time, we propose a predictive model that can classify and predict the execution of CFD settings. The model consists of two steps sequentially executing as seen in Fig. 3. The first step attempts to classify the execution ending status into two types of success or failure. While executing the Telemac-2D application, we observe that the application sometimes ends with failures due to lack of RAM. Hence, we utilize a decision tree algorithm, namely C4.5, to classify the execution ending status. In the second step, we attempt to predict the execution time of the settings whose statuses are successful. To address the task of prediction, in this study, we use Multi-layer perceptron (MLP) and a mixture of Multi-layer perceptrons.

4.1 C4.5

Among several Decision Tree algorithms, C4.5 is an extension of ID3 algorithm using the concept of information entropy. Instead of using Information Gain as ID3, C4.5 uses Gain Ratio to measure the entropy of attributes to identify a classified attribute [13]. The author also pointed out that C4.5 made a number of improvements to ID3. Although some improvements of C4.5 can handle with continuous attributes [14], the basic C4.5 algorithm is only suitable for types of data that nearly contain discrete attributes. The dataset that we collect satisfies this criterion, and thus C4.5 might be appropriate to our study. Moreover, one of the advantages of C4.5 algorithm is its rulersets that are grouped together

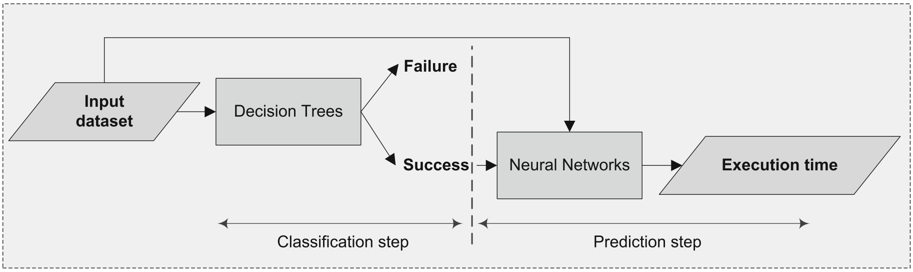


Fig. 3. The proposed model for predicting the execution time of the Telemac-2D application in the cloud computing system

in classes [18]. This characteristic of C4.5 makes the rulersets more easily be understood than the other algorithms.

4.2 Multi-layer Perceptron

In 1958, the first ANN was invented by psychologist Frank Rosenblatt [4]. Since then, there have been significant amounts of research that attempt to improve the performance of ANNs and apply ANNs to real-world problems [5]. These researchers on artificial neural networks (ANNs) were inspired by simulations of how the brain works in humans and other mammals [4,5]. The authors think of the human brain as a highly complex, nonlinear and parallel computer or information processing system capable of performing highly complex tasks. It is a fact that the brain is composed of cells called neurons. These neurons are responsible for performing complex computations as pattern recognition, perception or control. Typically, an artificial neural network is built up by a network of computing units, known as artificial neurons. These computing units are represented as nodes in the network and they are connected with each other through weights.

A Multi-layer Perceptron (MLP), a type of Multi-layer FeedForward Neural Network, consists of neurons whose activation functions are differentiable [5]. MLP has one or more hidden layers containing computation nodes. The computation nodes sometimes are called hidden neurons or hidden units. The task of these hidden units is to take part in the analysis of data between the input and output layers. By adding one or more hidden layers, the network can be capable of discovering many sophisticated relations between input and output of MLP.

To train MLP, in this study, we utilize a traditional learning algorithm called Back-Propagation (BP). The BP algorithm based on steepest descent method, was first published by Werbos in 1974 [17]. The BP algorithm has proved its effectiveness in several applications despite of some drawbacks, such as local convergence, over-fitting, and so on. Because the dataset used in this study is quite small (approximately 400 tuples) and have few attributes (5), we decide to utilize MLP to address the prediction of the execution time.

4.3 Mixture of Multi-layer Perceptrons

While observing the dataset, we realize that two parameters–the number of virtual machines and the number of processors–have complex and unobservable correlations with the execution time. Consequently, at the prediction step, when we apply only a MLP for learning the whole dataset, it takes a long time for the MLP to completely learn these correlations. Thus, to reduce the training time, we propose a model which is a mixture of MLPs, namely MiMLP. Figure 4 presents the structure of MiMLP. The idea of this model is quite simple. The original dataset is divided into several sub-datasets according to the number of virtual machines. The $1^{st}, 2^{nd}, ..., n^{th}$ sub-datasets are corresponding to sub-settings with $1, 2, ..., n$ virtual machines, respectively. Then, each MLP is trained with only one sub-dataset whose parameter of the number of virtual machines is removed. To predict for any setting, the model checks the number of virtual machines of this setting to decide which MLP is responsible for predicting the execution time of the setting.

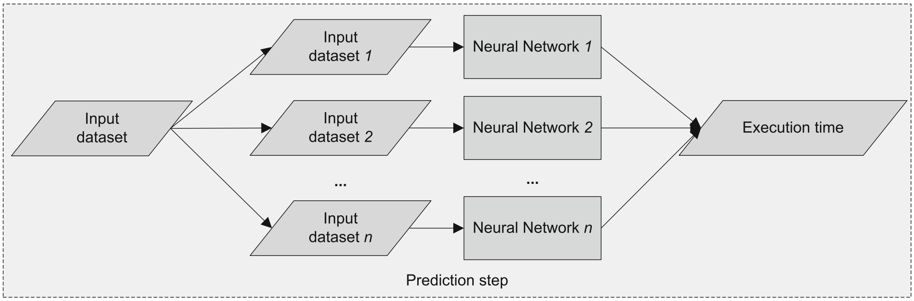


Fig. 4. The improvement of the proposed model at the prediction step, namely MiMLP

MiMLP works in such a way that it can be considered as a kind of ensemble learning, in particular, a bagging model without random factors.

5 Experimental Results

5.1 Classification Step

To assess the performance of the C4.5 algorithm in the classification step, we use the accuracy index which is defined as follows.

$$Accuracy = (TS + TF)/(TS + TF + FS + FF), \tag{1}$$

where TS is true success, TF is true failure, FS is false success, and FF is false failure.

Figure 5 presents the whole tree produced by the C4.5 algorithm. The testing step with 10 % of the dataset indicates that the accuracy of the C4.5 algorithm is

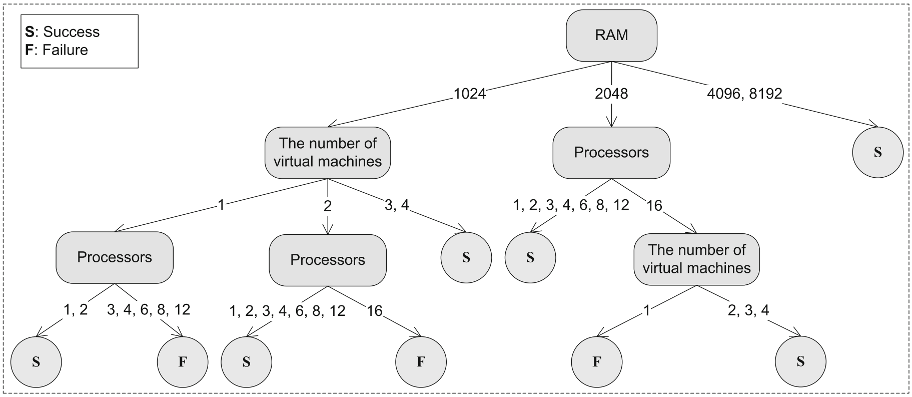


Fig. 5. The result tree of the C4.5 algorithm

almost 100 %. The high accuracy of the C4.5 algorithm can be easily understood since the main factor leading to failed endings of the CFD application is lack of RAM.

5.2 Prediction Step

The performance of MLP and MiMLP developed in this study are assessed by using various statistical performance evaluation criteria. The statistical measures considered are mean absolute relative error (MARE), coefficient of correlation (R), Nash-Sutcliffe coefficient (NS), and root mean squared error (RMSE).

$$MARE = \sum_{i=1}^{n} \frac{|O_i - P_i|}{O_i}, \tag{2}$$

$$R = \frac{\sum_{i=1}^{n}((O_i - \overline{O})(P_i - \overline{P}))}{\sqrt{\sum_{i=1}^{n}(O_i - \overline{O})^2}\sqrt{\sum_{i=1}^{n}(P_i - \overline{P})^2}}, \tag{3}$$

$$NS = 1 - \frac{\sum_{i=1}^{n}(O_i - P_i)^2}{\sum_{i=1}^{n}(O_i - \overline{O})^2}, \tag{4}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(O_i - P_i)^2}, \tag{5}$$

where O_i is the i^{th} observed value; $\overline{O}$ is the average observed value; P_i is the predicted value corresponding to i^{th} observed value; $\overline{P}$ is the average predicted value and n is the number of observed dataset.

We divide the dataset into a training sub-dataset and a testing sub-dataset. The sizes of the training sup-dataset and the testing sub-dataset are 300 and 100, respectively. The experimental results of two models are summarized in Table 2. Two model are appropriate to predict the execution time of the Telemac-2D

Table 2. The performance of MLP and MiMLP during training and testing phases

	MLP				MiMLP			
	MARE	R	NS	RMSE	MARE	R	NS	RMSE
Training	23.789	0.913	0.822	20.473	17.426	0.950	0.902	14.107
Testing	25.895	0.906	0.820	22.360	17.090	0.957	0.913	15.578

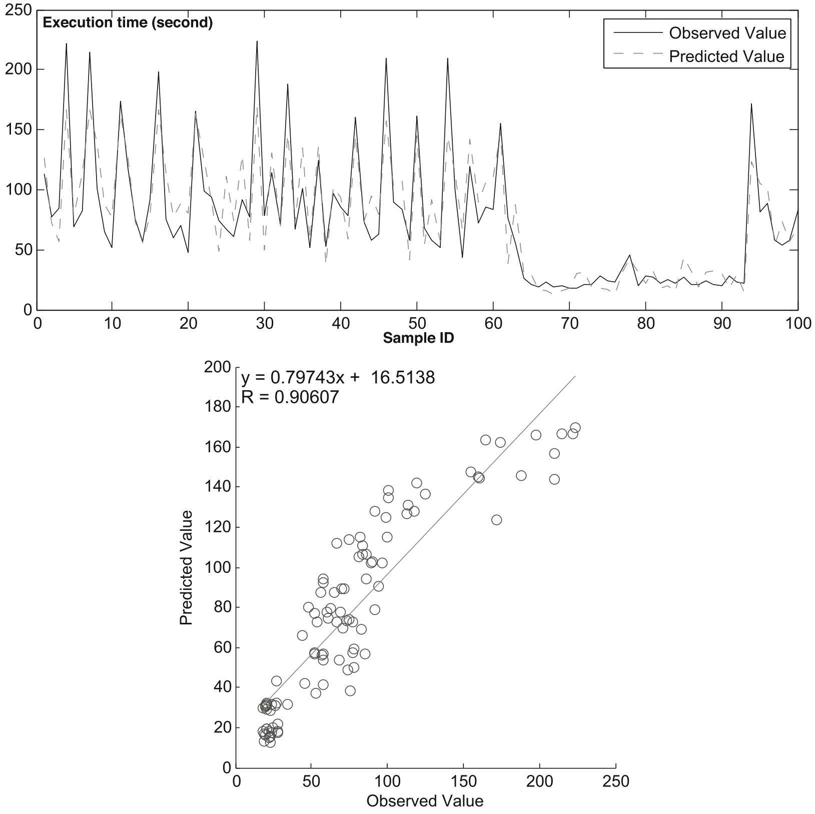


Fig. 6. The observed execution time and predicted values by MLP in testing phases

application since R, MARE of MLP and MiMLP are 0.906, 25.895 and 0.957, 17.090, respectively. However, with all better statistical measures, MiMLP is obviously superior to MLP. In addition, the experimental results indicate that the independent learning of MLPs on the sub-datasets is more effective than on the whole dataset. Figures 6 and 7 show the execution time and scatter plots of both the observed and the predicted values obtained by using MLP and MiMLP in the testing step, respectively. In Fig. 7, with R is 0.957, the predicted values by MiMLP are close to the observed values, and thus MiMLP can be used to predict the execution time of Telemac-2D application in the cloud computing system.

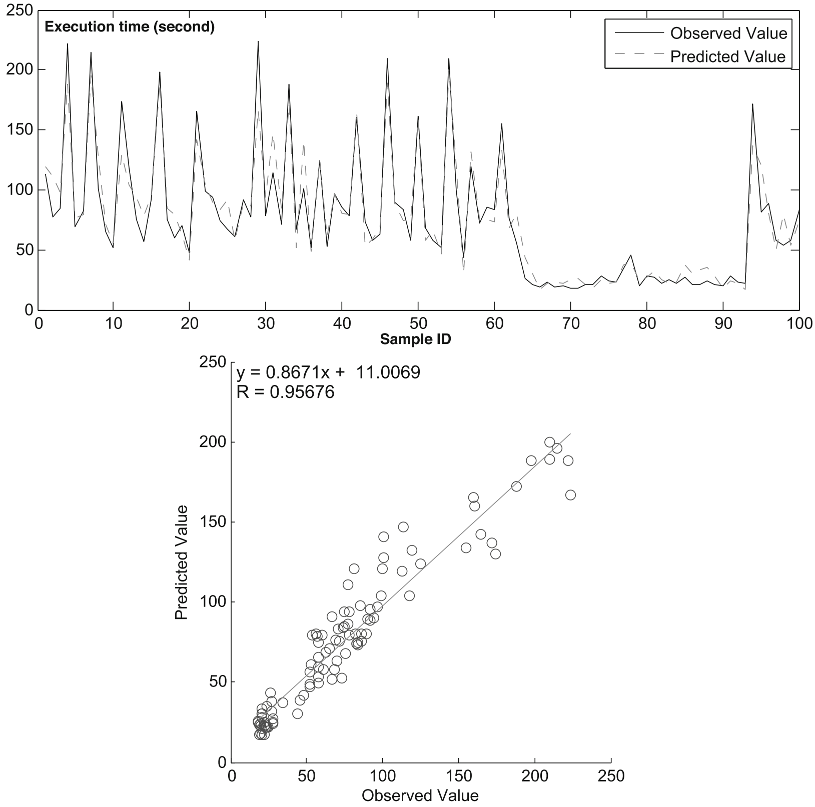


Fig. 7. The observed execution time and predicted values by MiMLP in testing phases

Table 3. The performance of the four MLPs of the MiMLP model during training and testing phases

MLPs	MARE	R	NS	RMSE
MLP corresponding to 1-virtual-machine settings	27.971	0.928	0.861	21.484
MLP corresponding to 2-virtual-machine settings	17.721	0.934	0.873	14.003
MLP corresponding to 3-virtual-machine settings	9.755	0.969	0.938	7.683
MLP corresponding to 4-virtual-machine settings	14.257	0.968	0.937	13.256
Averages	17.426	0.950	0.902	14.107

Note that the statistical measures of MiMLP in the train phase are the approximate averages of the statistical measures of all MLPs. The dataset consists of 1, 2, 3, and 4-virtual-machine settings, hence the MiMLP model contains four MLPs. The Table 3 describes the statistical measures of the four MLPs in the training phase.

6 Discussion and Conclusion

Estimating the execution time of a CFD application is very important to effectively schedule it in a cloud computing service. The machine learning-based model is used in this study to predict the execution time of the Telemac-2D application which simulates flood diversion in the MeKong Delta. The proposed model consists of two parts. The first part is to classify the execution ending status of the application by utilizing a decision tree algorithm called C4.5. In the second part, the execution time of the application is predicted by MLP and MiMLP. The experimental results indicate that the proposed model is appropriate since the accuracy of the C4.5 algorithm is 100 % and R and MARE of MiMLP are 0.957 and 17.090, respectively. In future work, the result of this study will be integrated into the whole process of scheduling in our clouding computing service.

Although the proposed approach orients to a specific CFD application, it is definitely applicable to other CFD applications. In practice, many CFD applications, like the Telemac-2D application, are often repeatedly executed with several different settings on a cloud-based high performance system. For another practical example, we are applying this approach for modeling the salinization of the MeKong Delta.

Acknowledgments. The authors would like to thank Faculty of Computer Science and Engineering, HCMC University of Technology for providing facilities for this study. The applications presented in this paper were tested on the High Performance Computing Center (HPCC) of the faculty. This research was funded by HCMC Department of Science and Technology, under contract number 39/2015/HD-SKHCN.

References

1. Anto, S., et al.: Stochastic based optimal resource provisioning in cloud computing. Int. J. Adv. Res. Comput. Eng. Technol. (IJARCET) **2**(2), 670 (2013)
2. Chaisiri, S., Lee, B.S., Niyato, D.: Optimization of resource provisioning cost in cloud computing. IEEE Trans. Serv. Comput. **5**(2), 164–177 (2012)
3. Doyle, J., Giotsas, V., Anam, M.A., Andreopoulos, Y.: Cloud instance management and resource prediction for computation-as-a-service platforms
4. Frank, R.: The perceptron: a probabilistic model for information storage and organization in the brain, cornell aeronautical laboratory. Psychol. Rev. **65**, 386–408 (1958)
5. Haykin, S.: Neural Networks and Learning Machines. Pearson International Edition, Upper Saddle River (2009)
6. Huang, L., Jia, J., Yu, B., Chun, B.G., Maniatis, P., Naik, M.: Predicting execution time of computer programs using sparse polynomial regression. In: Advances in Neural Information Processing Systems, pp. 883–891 (2010)
7. Ipek, E., de Supinski, B.R., Schulz, M., McKee, S.A.: An approach to performance prediction for parallel applications. In: Cunha, J.C., Medeiros, P.D. (eds.) Euro-Par 2005. LNCS, vol. 3648, pp. 196–205. Springer, Heidelberg (2005)
8. Jamshed, S.: Using HPC for Computational Fluid Dynamics: A Guide to High Performance Computing for CFD Engineers. Academic Press, Cambridge (2015)

9. Kasperkiewicz, J., Racz, J., Dubrawski, A.: HPC strength prediction using artificial neural network. J. Comput. Civ. Eng. **9**(4), 279–284 (1995)
10. Matsunaga, A., Fortes, J.A.: On the use of machine learning to predict the time and resources consumed by applications. In: Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, pp. 495–504. IEEE Computer Society (2010)
11. Oberkampf, W.L., Trucano, T.G.: Validation methodology in computational fluid dynamics. AIAA Pap. **2549**, 19–22 (2000)
12. Priya, R., de Souza, B.F., Rossi, A.L., de Carvalho, A.C.: Predicting execution time of machine learning tasks for scheduling. Int. J. Hybrid Intell. Syst. **10**(1), 23–32 (2013)
13. Quinlan, J.R.: Programs for Machine Learning. Morgan Kaufmann Publishers, Burlington (1993)
14. Quinlan, J.R.: Improved use of continuous attributes in c4.5. J. Artif. Intell. Res. **4**, 77–90 (1996)
15. Trucanoa, T.G., Swiler, L.P., Igusa, T., Oberkampf, W.L., Pilch, M.: Calibration, validation, and sensitivity analysis: whats what. Reliab. Eng. Syst. Saf. **91**, 1331–1357 (2006)
16. Thong, N., Loc, L.X., Tuan, H.D.: Impact assessment of flood diversion of Dong Thap Muoi for the Mekong Delta. Vietnam Water Resour. **9**, 3–12 (2015)
17. Werbos, P.: Beyond regression: new tools for prediction and analysis in the behavioral sciences, PhD. thesis. Harvard University, Cambridge (1974)
18. Wu, X., Kumar, V., Quinlan, J.R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Philip, S.Y., et al.: Top 10 algorithms in data mining. Knowl. Inf. Syst. **14**(1), 1–37 (2008)

TLCSim: A Large-Scale Two-Level Clustering Similarity Search with MapReduce

Trong Nhan Phan[1(✉)], Markus Jäger[1], Stefan Nadschläger[1],
Pablo Gómez-Pérez[1], Christian Huber[1], Josef Küng[1], and Cong An Nguyen[2]

[1] Faculty of Engineering and Natural Sciences (TNF),
Institute for Application Oriented Knowledge Processing (FAW),
Johannes Kepler University (JKU), Linz, Austria
{nphan,mjaeger,snadschlaeger,pgomez,chuber,jkueng}@faw.jku.at
[2] Dong Nai Social Insurance, Dong Nai, Vietnam
sisnetco@gmail.com

Abstract. Similarity search has become a principal operation not only in databases but also in diverse application domains. Very large datasets, however, pose a big challenge on its enormous volume-processing capability. In order to deal with the challenge, we propose a two-level clustering approach aiming at supporting fast similarity searches in massive datasets. In addition, we embed some pruning and filtering strategies into our methods so that redundancy-free data, data accuracy, inessential data accesses, unnecessary distance computations, and other following consequences are taken into account. Furthermore, we validate our methods by a series of empirical experiments in real big datasets. The results show that our approach performs better than the two inverted index-based approaches, especially when given big query batches.

Keywords: Similarity search · Scalability · Clustering · Filtering · Pruning · MapReduce · Hadoop

1 Introduction

The performance of similarity search has emerged as a persistent problem in which lots of effort has been engaging. Similarity search, on the one hand, is a non-trivial process since it is bound not only by I/O but also by CPU whilst distance computations in metric spaces are very expensive, especially time-intensive to evaluate similarity metrics between sets [26]. On the other hand, similarity search has to face a big challenge when there are large amounts of data. One popular approach to address this challenge is MapReduce paradigm [5], which aims at many large-scale computing problems. The basic idea is to divide a large problem into independent sub-problems which are then tackled in parallel by two tasks known as Map and Reduce.

The mechanism of MapReduce, however, is strictly bound by I/O since final key-value pairs must be written back to the distributed file system. In addition, it is worth noticing that MapReduce performs a full-scan manner that scans and

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 53–71, 2016.
DOI: 10.1007/978-3-319-48057-2_4

processes key-values pair by pair. For instance, assuming that there are three output files from a mapper and a reducer as seen in Fig. 1, the total files are 12 for four mappers and 9 for three reducers. In a single paradigm, reducers have to access and process all intermediate key-value pairs from 12 files emitted by mappers. Additionally, there is also an inevitable cost to combine these intermediate key-value pairs according to the key at the shuffling phase [5]. Besides, if the output of reducers is further used for another MapReduce operation, then the mappers of the latter MapReduce operation have to scan, split when necessary, and process all key-value pairs from 9 files output by the former MapReduce operation. Hence, the issue is that it is unnecessary to access and process all objects that are not relevant to a given object. Additionally, we observe that the reason for data redundancy initially comes from the output of mappers. From this point of view, there is a crucial need to minimize the number of inessential accesses as well as distance computations between other objects even though a state-of-the-art like MapReduce is applied to deal with large amounts of data.

Moreover, recent studies employing MapReduce to do similarity search with enormous data have at least two MapReduce cycles: one for pre-processing the input data beforehand; and the other for processing the query and deriving the final result [6,8,9,12–14,19,22]. We observe that if the input data are not well prepared in advance, the query processing will suffer more overheads because it has to run the data processing further for the related set of queries. Furthermore, we also notice that once the data input is well-pre-processed ahead of time, the first MapReduce cycle is run only once for arbitrary queries so that the second one can perform faster. This is actually useful in terms of application scenarios which do not need to re-access the original data input but to process given queries quickly. Thus, a need comes out to keep the best performance at the query-processing phase.

Motivated from these issues, we propose *TLCSim*, a large-scale two-level clustering similarity search, in order to achieve the aims. In our approach, we minimize examining unnecessary key-value pairs, which are illustrated as files with bold borders in Fig. 1, and sooner prune distance computations. As a result, our methods lead to not only reducing candidate size for reducers but also reaching other better consequences such as fewer computations, less I/O, less storage, and less shuffling cost when using MapReduce paradigm. In a nutshell, our main contributions are summed up as followings:

- We propose an element-based clustering scheme to organize data inside a cluster, which is built from the work in [17]. Besides, we attach our quick-pruning search strategies in order to support the proposed scheme so that it helps reduce inessential object accesses when given a query object.
- We propose our *TLCSim*, which applies the two-level clustering methods with MapReduce. Additionally, we organize the structure of key-value pairs and embed our filtering strategy during MapReduce processes.

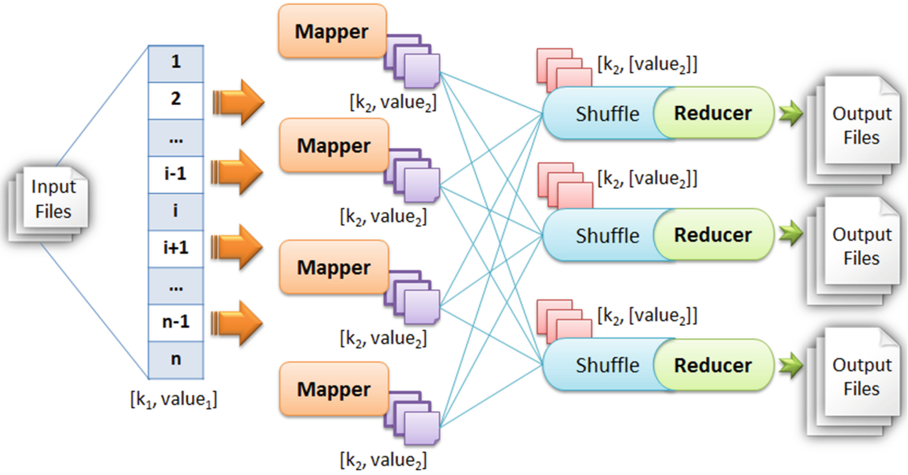


Fig. 1. MapReduce paradigm

– We intensively conduct empirical evaluations to validate our diverse methods provided by *TLCSim* with real big datasets and demonstrate their scalability. Furthermore, they are compared to the inverted index-based approaches from the work [13,15]. The results show that our work performs better than these approaches in terms of query processing, especially when given big query batches.

The rest of the paper is organized as follows. Section 2 introduces some key concepts that facility our approach. Next, Sect. 3 shows our related work. Then, we present our *TLCSim* in Sect. 4. Finally, we demonstrate our empirical experiments in Sect. 5 before our remarks in Sect. 6.

2 Preliminaries

2.1 Similarity Search

A workset or a corpus, denoted as Ω, consists of a set of document objects D$_i$, which is formally represented as $\Omega = \{D_1, D_2, D_3, \ldots, D_n\}$, and each document object D$_i$ is composed of a set of words as terms, which is shown as $D_i = \{term_1, term_2, term_3, \ldots, term_w\}$. Alternatively, K-shingles, defined as any sub-string having the length K found in the document, can be used to represent a document object [18,21]. As a consequence, a document object from now on is represented by a set of K-shingles such that $D_i = \{SH_1, SH_2, SH_3, \ldots, SH_k\}$, and the length of a document object $\|$D$_i$ $\|$ is known as the total number of shingles belonging to the object D$_i$.

An operation of similarity search is to look for similar objects compared to the given one so that their similarity scores should satisfy the pre-defined threshold. Our definition of similarity search is given as follows.

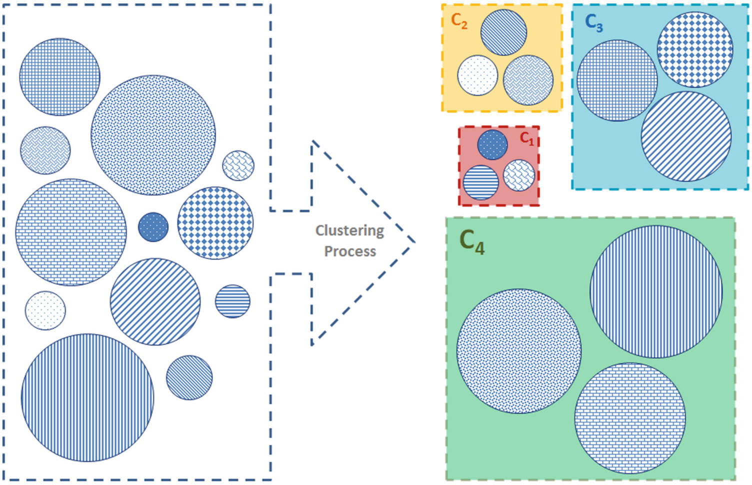


Fig. 2. Spiral clustering

Definition 2.1 (SIMILARITY SEARCH). *Given a query object D_q and a threshold δ, the similarity search looks for all document pairs (D_p, D_q) in the universal set Ω, such that their similarity scores SIM(D_p, D_q) $\geq \delta$.*

The value domain of SIM(D_p, D_q) is within the range [0, 1]. If the document D_p is more similar to the document D_q, their similarity score is close to 1. Otherwise, their similarity score is close to 0. In this paper, we illustrate our approach with the most widely-used similarity measure known as Jaccard coefficient, which is utilized for fast set-based similarity searches [14,15,18,19,25], such that $SIM(D_p, D_q) = \frac{\|D_p \cap D_q\|}{\|D_p \cup D_q\|}$. Last but not least, we use the sign $\mathbb{N}$ to point out a set of natural numbers, the sign [.] to demonstrate a list, the sign $[.]_{ord}$ denotes an ordered list, and the sign [[.], [.]] to specify a list of lists.

2.2 Spiral Clustering Scheme

Since length is an explicitly natural feature of a document object and we do not need to perform complex computations to obtain the length values, we employ this feature from our previous work [17] as a criterion to cluster documents in the corpus. Figure 2 illustrates our spiral clustering method, which clusters objects according to their length values. In other words, all objects in a same cluster have their length values satisfy the length parameter value as stated in Lemma 2.1 below.

Lemma 2.1 (SPIRAL CLUSTERING PROPERTY). *Let ξ be the set of clusters and λ be the pre-defined length parameter. Given any object D_i, which belongs to the cluster $C_n \in \xi$, the length NOS^i of the object D_i meets the following inequalities:*

$$[\lambda * (C_n - 1)] < NOS^i_n \leq (\lambda * C_n) \tag{1}$$

Basically, C_n stands for the cluster identifier, whose domain belongs to the natural number domain. Moreover, we do not have to evaluate every cluster and check every objects inside when given a query. In fact, we only need to check those inside the cluster bounds as stated in Lemma 2.2 below.

Lemma 2.2 (SPIRAL CLUSTER BOUNDS). *Let ξ be the set of n clusters, $\mathbb{N}$ be the set of natural numbers, C_{LB} be the lower bound cluster, C_{UB} be the upper bound cluster, λ be the pre-defined constant length parameter, δ be the similarity threshold, and Q be the query object. The candidate cluster bounds of Q, which is denoted by $\Xi(Q)$, is defined as $\{C_j \mid \forall j \in \mathbb{N}^*,\ C_j \in \xi \wedge C_{LB} \leq C_j \leq C_{UB}\}$, where:*

$$\begin{cases} C_{LB} = max((\frac{NOS^q}{\lambda} * \delta), 1) \\ C_{UB} = min((\frac{NOS^q}{\lambda * \delta}), n) \end{cases} \tag{2}$$

Moreover, the accuracy of doing similarity search by the spiral cluster bounds has been proved in our previous work [17]. In other words, real similar objects are included in the final result. Furthermore, our example below gives an illustration of how we identify spiral cluster bounds as specified in Lemma 2.2.

Example 2.1 (SPIRAL CLUSTER BOUNDS). *Let us suppose that the pre-defined constant length parameter λ = 800, a query object D_q with NOS^q = 2345, and the expected similarity threshold δ = 0.9. From Lemma 2.2, we have the spiral cluster bounds such as C_{LB} = 2.638125 and C_{UB} = 3.25694. As a consequence, we obtain n = 3, $\forall n \in \mathbb{N}^*$, and the candidate cluster bounds of D_q is finally identified as $\Xi(D_q) = \{C_3\}$.*

3 Related Work

Similarity search has called for much attention in improving its efficiency as well as its scalability. Many studies, however, do not employ any parallel mechanism or distributed computing to deal with big volumes of data [20,21,23,24,27]. Meanwhile, a few of them actually optimize parallel algorithms [1] or constructively interfere in parallel frameworks [2,7] to improve the performance of similarity search. Our approach is different from them such that we approach from the high level of schemes and employ the state-of-the-art paradigm known as MapReduce to achieve better performance for similarity search.

Doing similarity search with MapReduce has attracted lots of researches in order to deal with scalability. Deng et al. [6] present a MapReduce-based method for scalable string similarity joins with three phases in that the first MapReduce task is for filter stage and the last two MapReduce tasks are for verification stage. The method, however, needs to re-access the original datasets, together with the output from the first MapReduce, in the verification stage (i.e., the last two phases). Also aiming at string similarity join, Rong et al. [19] show their effort to achieve an efficient and scalable processing with three phases of MapReduce. Additionally, they apply multiple prefix filtering based on different global orderings, whose global information about the whole dataset cannot be easily accessed in a distributed environment. Moreover, the method performs a full-scan fashion that processes every object while our approach takes the advantage of clustering to prevent itself from accessing the unnecessary ones. On the other side, Zadeh and Goel [25] are interested in assessing whether a MapReduce algorithm is good or not. In their work, they point out that the performance of a MapReduce method mainly depends on the largest bucket to reduce and the size of intermediate key-value pairs at the shuffling phase. Thus, a crucial need to minimize the size of candidate size throughout MapReduce operations is risen, which also promotes us to meet the need. Meanwhile, Drew and Hahsler [8] introduce a word-based sequence classification scheme that uses MapReduce and Locality Sensitive Hashing for fast sequence comparison. In order to do that, they use two cycles of Map and four cycles of Reduce in total for the training and the classification phases. In reality such as in Hadoop, a Map task and a Reduce task cannot be independently separated. Hence, their method actually consumes four MapReduce cycles. Since the cost of a MapReduce operation is expensive, the more MapReduce cycles there are, the more costs we get. Lin [13] uses at least three MapReduce jobs for parallel queries method with the model "bag of words". Nevertheless, the size of the vocabulary histogram rapidly increases when data grow, which prevents the method from achieving high efficiency. Recently, Phan et al. [15] propose their inverted index-based approach in order to tackle with scalability in large data collections. Their method, however, is efficient with a single query. In our paper, we go on with an approach of document-based index, which is stated from our work [16] by how it overcomes the drawbacks of inverted index.

4 Our Approach

4.1 General Scheme

In order to avoid the dependence from both data sources and queries, we introduce our general scheme as illustrated in Fig. 3. Basically, the scheme consists of three independent consecutive work-flows known as data source, data index, and query running. The basic idea is to prepare data index from data sources in advance for further query processing. Because of high independence, each work-flow easily refreshes itself without violating any constraints with the others. Especially for query processing phase, we can now run multiple queries or query batches from a data index without re-accessing the original data sources. Furthermore, the fact that new data come to the data sources results in incrementally adding to the data index.

Finally, we employ two MapReduce jobs to support work-flow processing. Generally, our data-processing scheme is described by the two main steps as follows: (1) The first MapReduce job builds data indices from data sources; and (2) the second MapReduce job performs similarity search when given queries.

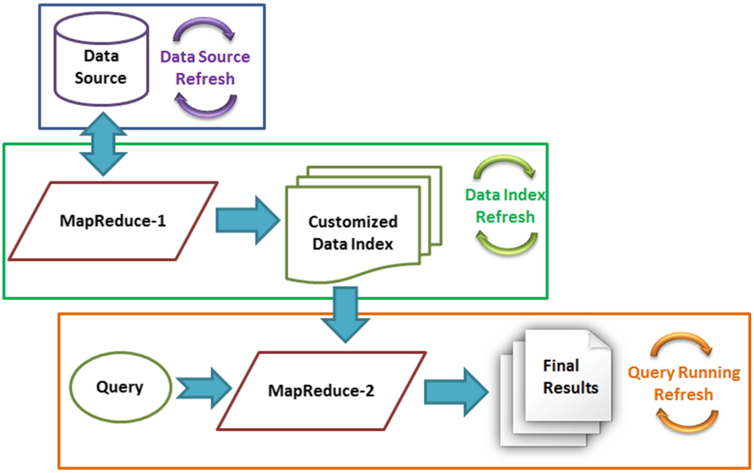


Fig. 3. Our general scheme

4.2 Element-Based Grouping

Once the candidate clusters have been identified from the spiral clustering scheme, we move one step further towards generating candidate pairs. Employing the form of inverted index as in the work [2,3,9,12–15,19], we perform grouping documents in their own cluster according to their own elements, which we choose and mention them as shingles. Hence, the element-based clustering scheme constitutes of the second level of clustering in our approach, and this process takes place in every cluster C$_n$. Basically, our element-based clustering scheme is defined as follows.

Definition 4.1 (ELEMENT-BASED CLUSTERING SCHEME). *Let ξ be the set of clusters and $C_n \in \xi$ be the considering cluster. For each element shingle SH_n^ρ in the cluster C_n, the element-based clustering process searches for objects D_i such as $\{D_i \mid D_i \in \Omega,\ SH_n^\rho \in D_i \wedge \rho = \bigcup i\}$.*

With the element-based clustering scheme, we avoid duplicate shingle checking in a cluster. When a shingle SH$_n^\rho$ is checked against that of the query object, we examine all possible objects D$_i$ that have SH$_n^\rho$ in common with the query. Our example below gives an illustration of how we identify candidate pairs from our element-based clustering scheme.

Example 4.1 (ELEMENT-BASED CLUSTERING SCHEME). *Figure 4 shows an example of the element-based clustering. In this example, we choose shingles as the elements to perform clustering inside a cluster. For instance, cluster C_n holds an element-based clustering structure as a list of list, which is denoted as $\{(SH_n^{248}, \{D_2, D_4, D_8\}), (SH_n^3, \{D_3\}), (\ldots, \{\ldots\}), (SH_n^{79}, \{D_7, D_9\})\}$. Suppose that a query object D_q also shares SH_n^{79} in the cluster C_n, our method checks SH_n^{79} only once and generate the two candidate pairs (D_7, D_q) and (D_9, D_q).*

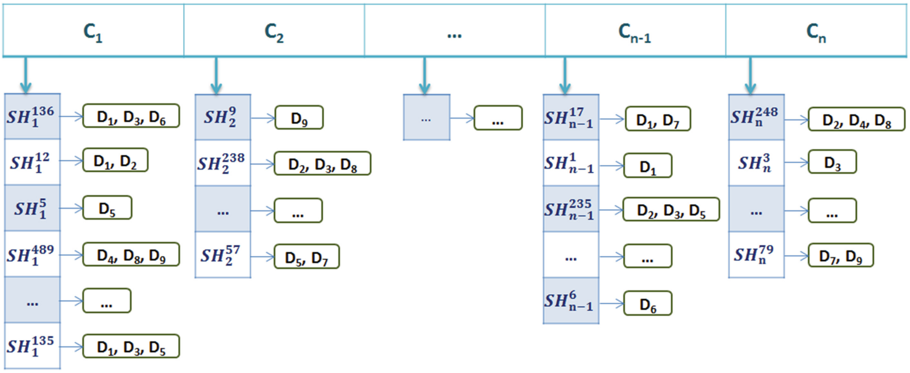


Fig. 4. Element-based clustering

4.3 Searching Strategies

Apart from avoiding multiple checking of the same shingle in a cluster, we also want to apply our quick pruning strategies so that the checking process will have the chance of being early terminated without processing unnecessary shingles. Considering most shingles in a cluster, we observe that the checking complexity in all-shingle search is O($\Im$) when the shingles are unordered. This also means all shingles in the cluster have to join the checking process, which leads to the case of redundancy problem.

In order to reduce the checking complexity, we firstly organize shingles in an ordered list. We let the sign u $\succeq$ v denote the greater string comparison between u and v while the sign u $\preceq$ v denote the smaller string comparison between u and v. Then we apply either of the two popular checking methods known as linear search [11] and binary search [4] as follows.

Definition 4.2 (LINEAR SEARCH). *Let ξ be the set of clusters, $C_n \in \xi$ be the considering cluster, and $L_n \in C_n$ is the ordered list of shingles, where $L_n = \{SH_n^1, SH_n^2, \ldots, SH_n^\rho\}$. For each element shingle SH_n^q shared by the query object D_q and the other objects D_i in the cluster C_n, the linear search process compares SH_n^q with the others SH_n^ρ in L_n and terminates when meeting the condition as $SH_n^\rho \succeq SH_n^q, \forall SH_n^\rho \in L_n$.*

With the linear search strategy, we have its complexity in the worst case and in the average case as stated in Lemma 4.2 below.

Lemma 4.2 (LINEAR SEARCH COMPLEXITY). *The linear search has the worst-case running time of $O(\rho)$ while it has the average-case running time of $O(\frac{\rho+1}{2})$.*

Proof. *The performance complexity of linear search is proved in [10].* □

Alternatively, we have the binary search defined as follows.

Definition 4.3 (BINARY SEARCH). *Let ξ be the set of clusters, $C_n \in \xi$ be the considering cluster, and $L_n \in C_n$ is the ordered list of shingles, where $L_n = \{SH_n^1, SH_n^2, \ldots, SH_n^\rho\}$. In addition, let $P_\rightarrow$ be the first index of L_n and $P_\leftarrow$ be the last index of L_n. For each element shingle SH_n^q shared by the query object D_q and the other objects D_i in the cluster C_n, the binary search process compares SH_n^q with the shingle L_n $[\Psi]$ at the middle-most point $\Psi = \lfloor \left(\frac{P_\rightarrow + P_\leftarrow}{2}\right) \rfloor$. If it does not match, the process recursively repeats its comparison with either the new first index $P_\rightarrow^{new}$ or the new last index $P_\leftarrow^{new}$ until it reaches its $\lfloor log_2 \, \| L_n \| \rfloor + 1$ probes at most as follows:*

$$\begin{cases} P_\leftarrow^{new} = \Psi - 1, & SH_n^q \preceq L_n[\Psi] \\ P_\rightarrow^{new} = \Psi + 1, & SH_n^q \succeq L_n[\Psi] \end{cases} \tag{3}$$

With the binary search strategy, we have its complexity in the worst case and in the average case as stated in Lemma 4.3 below.

Lemma 4.3 (BINARY SEARCH COMPLEXITY). *The binary search has both the worst-case and the average-case running times of $O(log_2 \; \rho)$.*

Proof. *The performance complexity of binary search is proved in [10].* □

4.4 TLCSim with MapReduce

In this section, we introduce our *TLCSim*, known as ***T***wo-***L***evel ***C***lustering ***Sim***ilarity search. In addition, we equip *TLCSim* with MapReduce paradigm to intensively facilitate its large-scale data processing. *TLCSim* consists of two main phases as follows: (1) Clustering phase; and (2) Similarity search phase. Each phase corresponds to a MapReduce cycle which includes one Map task and one Reduce task. Table 1 presents the overview of MapReduce operations. More details are given in the followings.

Table 1. The overview of MapReduce operations

TASK	**INPUT FORM**	**OUTPUT FORM**
MAP-1	[D$_i$]	[URL$_i$, NOS$_{ix}$@[SH$_k$]]
REDUCE-1	[URL$_i$, NOS$_{ix}$@[SH$_k$]]	[C$_{ID}$, [SH$_k$!@![URL$_i$@NOS$_i$]]$_{ord}$]
MAP-2	[C$_{ID}$, [SH$_k$!@![URL$_i$@NOS$_i$]]$_{ord}$]	[URL$_i$-URL$_j$@NOS$_i$@NOS$_j$, 1]
REDUCE-2	[URL$_i$-URL$_j$@NOS$_i$@NOS$_j$, 1]	[URL$_i$-URL$_j$, SIM$_{ij}$]

ACRONYMS

- D$_i$: a document object
- SH$_k$: a shingle
- URL$_i$ or URL$_j$: an uniform resource locator of D$_i$ or D$_j$
- NOS$_{ix}$: the partial number of shingles of D$_i$
- NOS$_i$ or NOS$_j$: the total number of shingles of D$_i$ or D$_j$
- C$_{ID}$: a cluster identification
- SIM$_{ij}$: the similarity score between D$_i$ and D$_j$
- Some special symbols such as "@" and "!@!" are used to separate the values

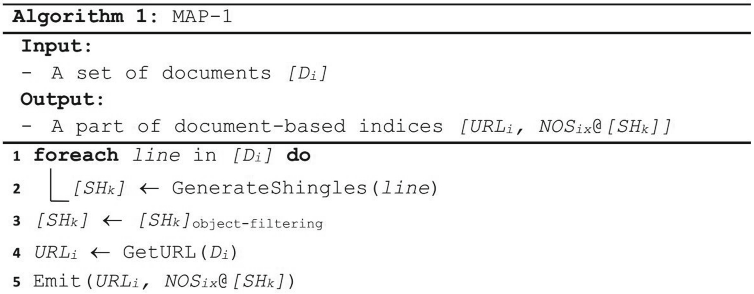
Clustering Phase. The goal of the clustering phase is to group similar objects before examining them against a query object. In order to do that, the data input is at first extracted to shingles by the mappers at MAP-1 task. In other words, these mappers will emit the intermediate key-value pairs of the form $[URL_i, NOS_{ix}@[SH_k]]$. It is worth noticing that the number of shingles of each document (i.e., NOS$_{ix}$) obtained from what mappers emit may be just a partial quantity. The reason behind is that the data input will be partitioned into smaller chunks if its size is over the capacity of the cluster of commodity machines. Moreover, each chunk may be processed by different mappers in different machines. Besides, we use URL$_i$ to identify the locator of the document D$_i$ in a distributed system. According to the formula (1), each URL$_i$ is then clustered by its total NOS$_i$ in comparison with the length parameter to form the first level of clustering. How to generate the second level of clustering, on the other hand, relies on the element-based clustering scheme. The reducers at REDUCE-1 task, therefore, aggregate the partial quantities in order to acquire the full length of a document. Next, the reducers at REDUCE-1 task organize their data according to the element-based clustering. Besides, the data are properly sorted so that we can employ the quick pruning strategy and avoid redundancy. As a consequence, the final output released from the reducers at REDUCE-1 task has the form of $[C_{ID}, [SH_k!@![URL_i@NOS_i]]_{ord}]$.

Similarity Search Phase. After the clustering phase, we have our data prepared in the two-level clusters from particular datasets. The similarity search phase then searches similar objects over the two-level clusters within one MapReduce cycle. In our approach, *TLCSim* performs an exact similarity search. It firstly performs checking candidate clusters according to the Lemma 2.2. Once a candidate cluster is hold, the similarity search phase continually performs quick pruning strategies to generate candidate pairs. The mappers at MAP-2 task, therefore, emit the intermediate key-value pairs of the form $[URL_i - URL_j@NOS_i@NOS_j, 1]$, which will be pulled to the reducers at REDUCE-2 task to actually compute similarity scores. Before outputting the final result of the form $[URL_i - URL_j, SIM_{ij}]$, it is necessary for the reducers at REDUCE-2 task to verify against the query filtering such as the similarity threshold, for example, so that we can avoid false positive, which indicates the case where dissimilar objects are actually included in the final result.

In overall, we construct the form of the intermediate key-value pairs as the form of what we call a document-based index [16] instead of the form of an inverted index because of two main reasons as follows. Firstly, it is naturally convenient to derive the number of shingles of each document to meet the need of length-based clustering. Secondly, acquiring the total number of shingles of each document is important not only for the similarity computing itself but also for other fast set-based similarity computing. Furthermore, filtering techniques are employed at MAP-1 task in order to not only discard duplicates that contribute nothing to the similarity scores but also refine special symbols emerging in shingles. It is from natural language processing but necessary in our approach so that similar shingles in terms of either vocabulary or meaning should be treated as one. As a consequence, doing this way reinforces the accuracy when computing similarity with regardless of its bit cost. Last but not least, another goal of our approach is to minimize the cost of running query batches when it is experienced better by the empirical experiments given in Sect. 5.

4.5 Algorithms

In this section, we provide our algorithms in the form of pseudo-codes. In general, TLCSim takes two MapReduce jobs. The first MapReduce job is to build indices with respect to the our schemes in Sect. 4 whereas the second MapReduce job is to do similarity search over the indices.



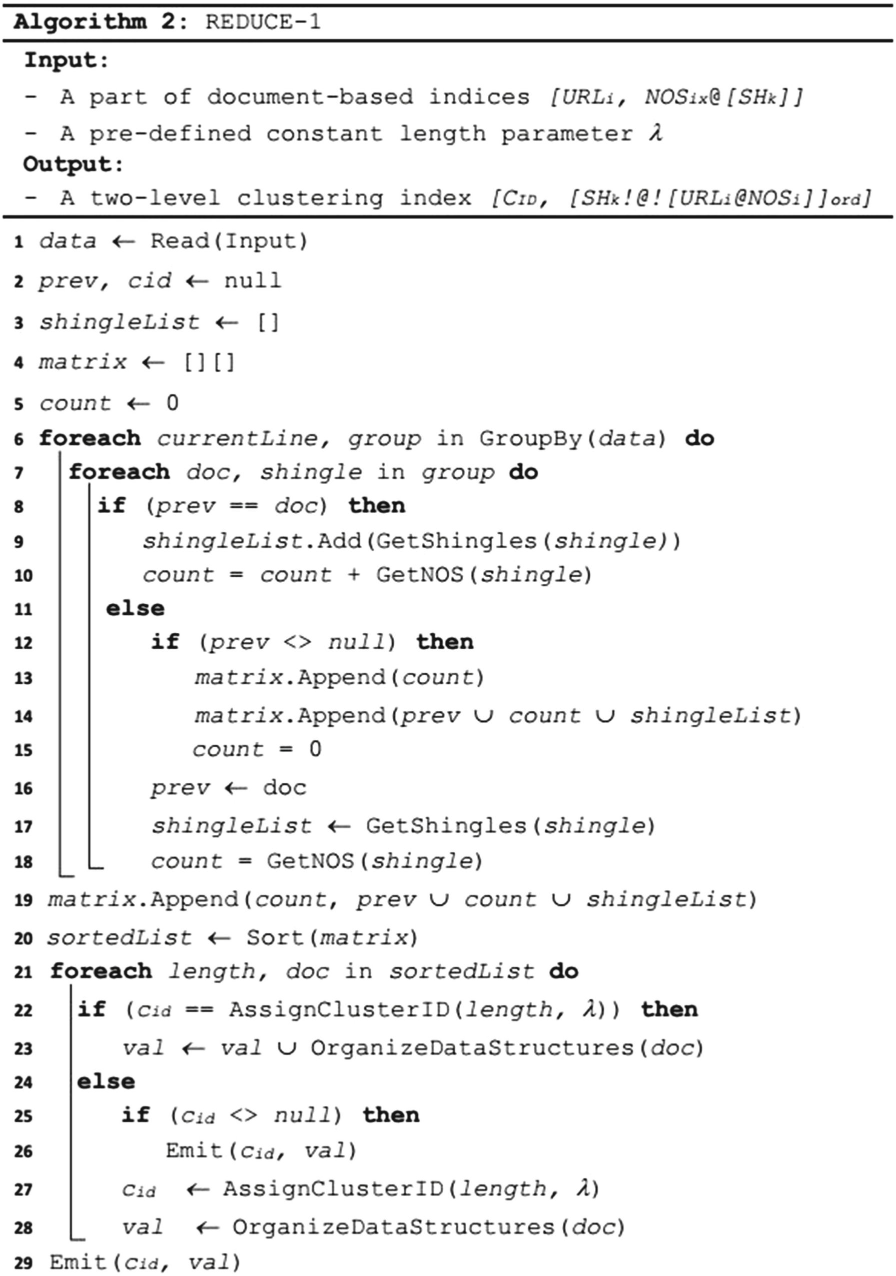
```
Algorithm 1: MAP-1
Input:
- A set of documents [$D_i$]
Output:
- A part of document-based indices [$URL_i$, $NOS_{ix}$@[$SH_k$]]
1 foreach line in [$D_i$] do
2     [$SH_k$] ← GenerateShingles(line)
3 [$SH_k$] ← [$SH_k$]$_{\text{object-filtering}}$
4 $URL_i$ ← GetURL($D_i$)
5 Emit($URL_i$, $NOS_{ix}$@[$SH_k$])
```

Fig. 5. MAP-1 algorithm

Figure 5 introduces our MAP-1 algorithm. The mappers at MAP-1 task parse documents D_i and then generate shingles from their content, which can be seen as in lines 1–2. Next, we perform object-filtering on the shingle set as in line 3. Then, the mappers get URLs from documents as in line 4 before emitting a part of document-based indices as in line 5.

Figure 6 shows our REDUCE-1 algorithm. This is the phase where we organize our data with respect to the spiral clustering scheme in Sect. 2.2 and the element-based clustering scheme in Sect. 4.2 when given the length parameter λ. More specifically, the reducers at REDUCE-1 task read their input as in line 1. Next, necessary variables are initialized as in lines 2–5. Then, documents and their shingles are collected into a two dimensional matrix as in lines 6–19. Basically, the matrix manages information about the documents such as their URLs, shingles, and total number of shingles. After that, the matrix is sorted by the total number of shingles as in line 20. Finally, the documents are grouped into corresponding clusters as in lines 21–29.

Figure 7 illustrates our MAP-2 algorithm. Firstly, the mappers at MAP-2 task read their input as in line 1. Secondly, they extract the information such as shingles, total number of shingles, and URL from the given query as in lines 2–4. Thirdly, they compute the spiral cluster bounds as in lines 5–6 when given a similarity threshold δ. Fourthly, the mappers consider those clusters that are in the range of the spiral cluster bounds for similarity search as in lines 7–8. After that, the mappers start looking for any overlap between the shingles of



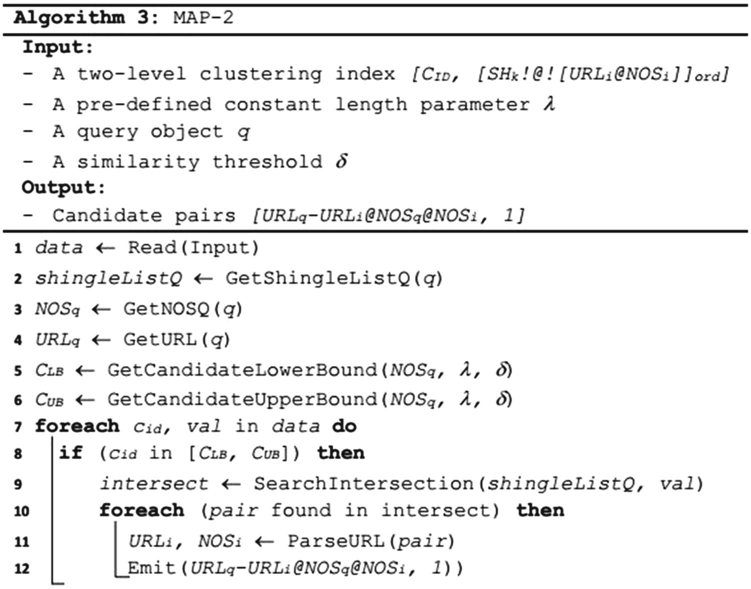
```
Algorithm 2: REDUCE-1
Input:
- A part of document-based indices [URL$_i$, NOS$_{ix}$@[SH$_k$]]
- A pre-defined constant length parameter $\lambda$
Output:
- A two-level clustering index [C$_{ID}$, [SH$_k$!@![URL$_i$@NOS$_i$]]$_{ord}$]

data ← Read(Input)
prev, cid ← null
shingleList ← []
matrix ← [][]
count ← 0
foreach currentLine, group in GroupBy(data) do
    foreach doc, shingle in group do
        if (prev == doc) then
            shingleList.Add(GetShingles(shingle))
            count = count + GetNOS(shingle)
        else
            if (prev <> null) then
                matrix.Append(count)
                matrix.Append(prev ∪ count ∪ shingleList)
                count = 0
            prev ← doc
            shingleList ← GetShingles(shingle)
            count = GetNOS(shingle)
matrix.Append(count, prev ∪ count ∪ shingleList)
sortedList ← Sort(matrix)
foreach length, doc in sortedList do
    if (C$_{id}$ == AssignClusterID(length, $\lambda$)) then
        val ← val ∪ OrganizeDataStructures(doc)
    else
        if (C$_{id}$ <> null) then
            Emit(C$_{id}$, val)
        C$_{id}$ ← AssignClusterID(length, $\lambda$)
        val ← OrganizeDataStructures(doc)
Emit(C$_{id}$, val)
```

Fig. 6. REDUCE-1 algorithm

documents in those clusters and those of the given query as in line 9. For each document found in the intersection, the mappers then emit candidate pairs as in lines 10–12.

Figure 8 visualizes our REDUCE-2 algorithm. More concretely, the reducers at REDUCE-2 task read their input as in line 1. Next, necessary variables are initialized as in lines 2–3. Finally, the reducers aggregate the candidate pairs and compute their similarity scores as in lines 4–15.



```
Algorithm 3: MAP-2
 Input:
 - A two-level clustering index [C_ID, [SH_k!@![URL_i@NOS_i]]_ord]
 - A pre-defined constant length parameter $\lambda$
 - A query object $q$
 - A similarity threshold $\delta$
 Output:
 - Candidate pairs [URL_q-URL_i@NOS_q@NOS_i, 1]
1  data ← Read(Input)
2  shingleListQ ← GetShingleListQ($q$)
3  NOS_q ← GetNOSQ($q$)
4  URL_q ← GetURL($q$)
5  C_LB ← GetCandidateLowerBound(NOS_q, $\lambda$, $\delta$)
6  C_UB ← GetCandidateUpperBound(NOS_q, $\lambda$, $\delta$)
7  foreach c_id, val in data do
8      if (c_id in [C_LB, C_UB]) then
9          intersect ← SearchIntersection(shingleListQ, val)
10         foreach (pair found in intersect) then
11             URL_i, NOS_i ← ParseURL(pair)
12             Emit(URL_q-URL_i@NOS_q@NOS_i, 1))
```

Fig. 7. MAP-2 algorithm

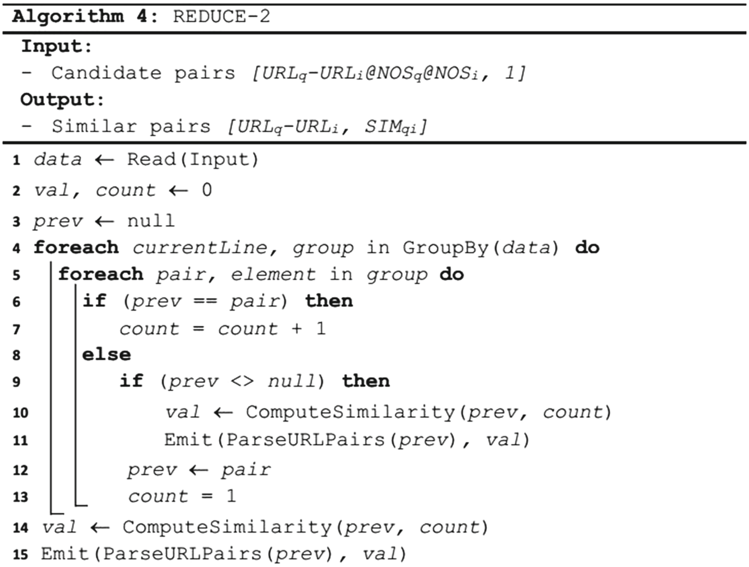
5 Empirical Experiments

5.1 Environment Settings

In our experiments, we deploy the stable version 1.2.1 of Hadoop[1] on the cluster of commodity machines named Alex[2], which has 48 nodes and 8 CPU cores and either 96 or 48 GB RAM for each node. Besides, the number of reducers for a reduce operation is set to 168. In the meantime, the possible heap size of the cluster is about 889 MB, and each file in Hadoop Distributed File System has 64 MB Block Size. Normally, we leave other Hadoop configurations in default mode as much as possible, for we want to keep the most initial settings which a cluster of commodity machines may get although some parameters can be tuned or optimized to fit the Alex cluster. Moreover, each benchmark has its fresh running. Last but not least, all the experiments for one type of query are consecutively run so that their environments are close as much as possible.

[1] http://hadoop.apache.org/docs/r1.2.1/mapred-default.html.

[2] http://www.jku.at/content/e213/e174/e167/e186534.



```
Algorithm 4: REDUCE-2
Input:
- Candidate pairs [$URL_q$-$URL_i$@$NOS_q$@$NOS_i$, 1]
Output:
- Similar pairs [$URL_q$-$URL_i$, $SIM_{qi}$]

data ← Read(Input)
val, count ← 0
prev ← null
foreach currentLine, group in GroupBy(data) do
    foreach pair, element in group do
        if (prev == pair) then
            count = count + 1
        else
            if (prev <> null) then
                val ← ComputeSimilarity(prev, count)
                Emit(ParseURLPairs(prev), val)
            prev ← pair
            count = 1
val ← ComputeSimilarity(prev, count)
Emit(ParseURLPairs(prev), val)
```

Fig. 8. REDUCE-2 algorithm

5.2 Datasets

We use Gutenberg datasets[3], a large single collection of free electronic eBooks, which are searched for their similarity together with experience for large text files. Moreover, the datasets are separately organized into five data packages including 3000 files, 6000 files, 9000 files, 12000 files, and 15000 files. These files randomly selected from the Gutenberg repository have their sizes ranging from 1 KB to 252 KB. Besides, we also organize smaller data packages including 50 files, 100 files, 150 files, and 200 files so that we are able to experience the case of overloaded memory. Furthermore, a query batch consists of a set of queries in that its cardinality depends on the number of single query needing to be processed once for similarity at the same running time.

5.3 Method Comparison

For our comparison experiments, we compare *TLCSim* methods and inverted index-based methods as following:

- *The Parallel Queries method (PQ)*: shows an inverted-index based method with the model “bag of words” as described in [13]. Moreover, we refer the method as *PQ* when considering only the cost of the last two MapReduce jobs. Otherwise, we mention it as *PQ Extended (PQE)* adding the cost of the first MapReduce job for building the histogram of vocabulary.

[3] http://www.gutenberg.org/.

- *Inverted_Index*: denotes an inverted-index based method in [15], which is shown as a fast similarity search that performs better than the base-line inverted index method in a large dataset.
- *TLCSim_WOO*: refers to our method when the second-level of clustering is not sorted. Consequently, the all-shingle search is conducted to generate candidate pairs.
- *TLCSim_WO*: indicates our method when the second-level of clustering is sorted. In addition, the linear search is taken to generate candidate pairs.
- *TLCSim_BS*: demonstrates our method when the second-level of clustering is sorted. Besides, the binary search is performed to generate candidate pairs.

Moreover, we have implemented these methods with MapReduce (MR) in Python and make them run by Hadoop streaming[4], which is a utility that comes with the Hadoop distribution. These methods other than *PQE* take two MapReduce cycles, which is called Total MR including MR-1 and MR-2.

5.4 Evaluation

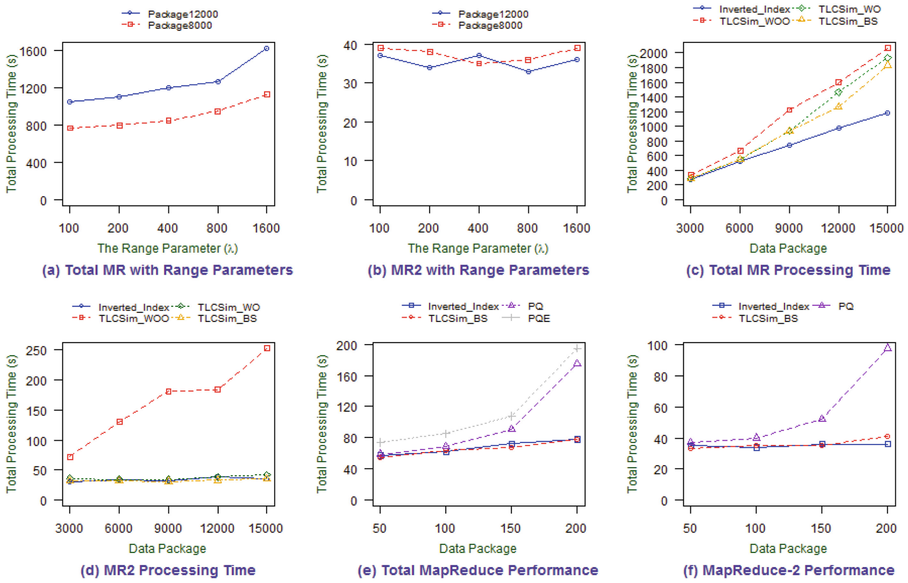
In our first experiments, we examine the influence of the range parameter in our methods with a single query. It is exponentially tuned from 100 to 3200 and is examined with the data packages 8000 and 12000. Due to the fact that the values of MR-2 are much smaller than that of MR-1, we separately visualize them into two different line charts.

In Fig. 9a, we observe that the total processing times of Total MR slightly rise when increases from 100 to 800 and sharply rise when increases from 800 to 1600. The performances of MR-2, however, are different from those of Total MR in Fig. 9b, and the best ones are reached when is set to 400 and 800, correspondingly. Consequently, λ is set to 800 for the other experiments after being considered in terms of both performance and the large dataset size.

For the upcoming experiments, we measure the performance of the candidate methods with a single query. Overall, *Inverted_Index* performs better whilst *TLCSim_WOO* performs worse than the others as in Fig. 9c. With the packages 3000 and 6000, the performances of *TLCSim_WO*, *TLCSim_BS*, and *Inverted_Index* are not much different. Nevertheless, there are big gaps with the data packages 9000, 12000, and 15000. The average gap rate between *Inverted_Index* and *TLCSim* is 35 %. On the other side, Fig. 9d illustrates MR2 processing time among the methods. Generally, the performance of *TLCSim_WOO* is the worst and sharply rises. In contrast, the other methods outperform *TLCSim_WOO* and perform closely to one another.

Next, we compare our method representative *TLCSim_BS* with *Inverted_Index*, *PQ*, and *PQE*. It is not surprise that both *PQ* and *PQE* consume the most time for the whole process, which is illustrated in Fig. 9e. The reason behind is that *PQE* takes more MapReduce jobs than the others. In the meantime, *PQ* uses more computations while they depend on the size of

[4] http://hadoop.apache.org/docs/r1.2.1/streaming.html.



(a) Total MR with Range Parameters

(b) MR2 with Range Parameters

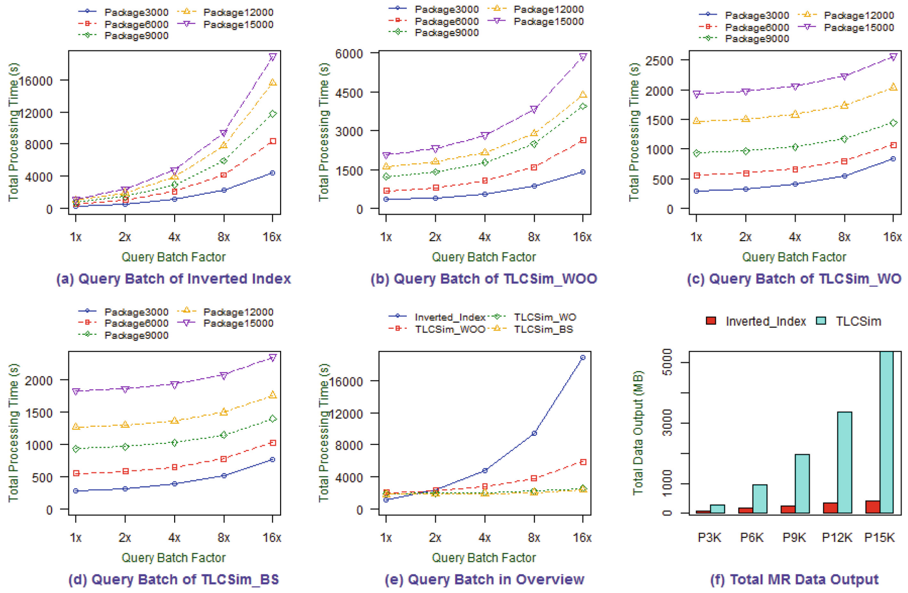
(c) Total MR Processing Time

(d) MR2 Processing Time

(e) Total MapReduce Performance

(f) MapReduce-2 Performance

Fig. 9. Total MapReduce processing time with a single query



(a) Query Batch of Inverted Index

(b) Query Batch of TLCSim_WOO

(c) Query Batch of TLCSim_WO

(d) Query Batch of TLCSim_BS

(e) Query Batch in Overview

(f) Total MR Data Output

Fig. 10. Total MapReduce processing time with query batches

vocabulary histogram. As a consequence, *PQ* is not efficient in terms of query processing. As illustrated in Fig. 9f, *PQ* takes the most time to process the given query. Even worse, the gap between *PQ* and the other methods rapidly rises when the dataset size changes from 100 to 200. More specifically, the total MR gap between *PQE* and *TLCSim_BS* increases from 25.68 % with 50 files to 60 % with 200 files. Moreover, the MR-2 gap between *PQ* and *TLCSim_BS* rises from 7.27 % with 50 files to 124.36 % with 200 files. Meanwhile, *TLCSim_BS* and *Inverted_Index* tend to have not much difference in their performances.

In the following, we conduct our essential experiments with query batches in Fig. 10. Each batch consists of a set of queries that need to be processed once for similarity at the same running time. There are five query batches that are exponentially set from 1x to 16x. With these query batches, Fig. 10a visualizes the similarity performances of *Inverted_Index*, Fig. 10b indicates that of *TLCSim_WOO*, Fig. 10c shows that of *TLCSim_WO*, and Fig. 10d presents that of *TLCSim_BS*. In general, the total processing times of *Inverted_Index* exponentially increase while that of *TLCSim* gradually grow. This trend also indicates that the query batch sizes cause bigger impacts on the total processing times than the dataset sizes. As a consequence, *TLCSim* methods moderately takes their processing time while *Inverted_Index* sharply does when the sizes of query batches exponentially increase.

Besides, Fig. 10e shows the overview of performance relevance among the methods with regard to the number of query batches. Intuitively, there are no much differences about the performances of *TLCSim_WO* and *TLCSim_BS* whilst there are very big gaps about the performances of *Inverted_Index* compared to the others. As a result, the performance of *TLCSim* highly outperforms that of *Inverted_Index* within the rates from 21.05 % to 88.68 %. Eventually, *TLCSim* methods produce mostly 84.91 % output more than *Inverted_Index* does after MR1 as seen in Fig. 10f. The reason is that *TLCSim* still preserves most of the input data at this phase whereas *Inverted_Index* earlier starts filtering the input data against the given query.

To sum up, though *TLCSim* methods take more time to prepare data than *Inverted_Index* at the clustering phase, the total query processing time of *TLCSim_WO* and *TLCSim_BS* at the similarity search phase is very close to that of *Inverted_Index* when given a query. Moreover, the overall performance of *TLCSim* definitely outperforms that of *Inverted_Index* in terms of query batches.

6 Summary

In this paper, we propose TLCSim, a large-scale two-level clustering similarity search with MapReduce. In parallel, we equip our methods with quick-pruning and filtering strategies so that we can improve the performance as well as the accuracy when computing similarity scores. Moreover, we manage empirical evaluations to validate our proposed methods with real large datasets. The results show that our approach supports fast similarity searches in massive datasets better than the inverted index-based approaches, especially with a batch of queries.

For our future work, we find it interesting to conduct more empirical experiments with other related work in that they may have different approaches and methods. Moreover, similarity queries would also be taken into account througout the comparisons.

Acknowledgments. Our sincere thanks to Mr. Faruk Kujundi, Scientific Computing, Information Management team, Johannes Kepler University Linz, for kindly supporting us with Alex Cluster.

References

1. Alabduljalil, M.A., Tang, X., Yang, T.: Optimizing parallel algorithms for all pairs similarity search. In: Proceedings of the 6th ACM WSDM, pp. 203–212. ACM, New York (2013)
2. Baraglia, R., De Francisci, M., Lucchese, C.: Document similarity self-join with MapReduce. In: Proceedings of the 2010 IEEE ICDM, pp. 731–736. IEEE Computer Society, Washington (2010)
3. Bayardo, R.J., Ma, Y., Srikant, R.: Scaling up all pairs similarity search. In: Proceedings of the 16th WWW, pp. 131–140. ACM, New York (2007)
4. Cormen, T.H., Stein, C., Rivest, R.L., Leiserson, C.E.: Introduction to Algorithms. McGraw-Hill Higher Education, New York (2001)
5. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. Commun. ACM **51**(1), 107–113 (2008)
6. Deng, D., Li, G., Hao, S., Wang, J., Feng, J.: MassJoin: a MapReduce-based method for scalable string similarity joins. In: 30th IEEE ICDE, pp. 340–351 (2014)
7. Dittrich, J., Richter, S., Schuh, S., Quian-Ruiz, J.-A.: Efficient or Hadoop: why not both? IEEE Data Eng. Bull. **36**(1), 15–23 (2013)
8. Drew, J., Hahsler, M.: Strand: fast sequence comparison using MapReduce and locality sensitive hashing. In: Proceedings of the 5th ACM BCB, pp. 506–513. ACM, New York (2014)
9. Elsayed, T., Lin, J., Oard, D.W.: Pairwise document similarity in large collections with MapReduce. In: Proceedings of the 46th ACL-HLT: Short Papers, pp. 265–268. Association for Computational Linguistics, Stroudsburg (2008)
10. Jenkyns, T., Stephenson, B.: Fundamentals of Discrete Math for Computer Science: A Problem-Solving Primer. Springer, London (2012)
11. Knuth, D.E.: The Art of Computer Programming. Sorting and Searching, vol. 3, 2nd edn. Addison Wesley Longman Publishing Co. Inc., Boston (1998)
12. Li, R., Ju, L., Peng, Z., Yu, Z., Wang, C.: Batch text similarity search with MapReduce. In: Du, X., Fan, W., Wang, J., Peng, Z., Sharaf, M.A. (eds.) APWeb 2011. LNCS, vol. 6612, pp. 412–423. Springer, Heidelberg (2011). doi:10.1007/978-3-642-20291-9_46
13. Lin, J.: Brute force and indexed approaches to pairwise document similarity comparisons with MapReduce. In: Proceedings of the 32nd ACM SIGIR, pp. 155–162. ACM, New York (2009)
14. Metwally, A., Faloutsos, C.: V-SMART-Join: a scalable MapReduce framework for all-pair similarity joins of multisets and vectors. Proc. VLDB Endowment **5**(8), 704–715 (2012)
15. Phan, T.N., Küng, J., Dang, T.K.: An elastic approximate similarity search in very large datasets with MapReduce. In: Hameurlain, A., Dang, T.K., Morvan, F. (eds.) Globe 2014. LNCS, vol. 8648, pp. 49–60. Springer, Heidelberg (2014)

16. Phan, T.N., Jäger, M., Nadschläger, S., Küng, J., Dang, T.K.: An efficient document indexing-based similarity search in large datasets. In: Proceedings of the 2nd FDSE, pp. 16–31 (2015)
17. Phan, T.N., Küng, J., Dang, T.K.: eHSim: an efficient hybrid similarity search with MapReduce. In: Proceedings of the 30th IEEE AINA, pp. 422–429. IEEE Computer Society (2016)
18. Rajaraman, A., Ullman, J.D.: Chapter 3: finding similar items. In: Mining of Massive Datasets, pp. 71–127. Cambridge University Press (2011)
19. Rong, C., Lu, W., Wang, X., Du, X., Chen, Y., Tung, A.K.H.: Efficient and scalable processing of string similarity join. IEEE TKDE **25**(10), 2217–2230 (2013)
20. Satuluri, V., Parthasarathy, S.: Bayesian locality sensitive hashing for fast similarity search. Proc. VLDB Endowment **5**(5), 430–441 (2012)
21. Theobald, M., Siddharth, J., Paepcke, A.: SpotSigs: robust and efficient near duplicate detection in large web collections. In: Proceedings of the 31st ACM SIGIR, pp. 563–570. ACM, New York (2008)
22. Vernica, R., Carey, M.J., Li, C.: Efficient parallel set-similarity joins using MapReduce. In: Proceedings of the 2010 ACM SIGMOD, pp. 495–506. ACM, New York (2010)
23. Wang, J., Li, G., Deng, D., Zhang, Y., Feng, J.: Two birds with one stone: an efficient hierarchical framework for top-k and threshold-based string similarity search. In: Gehrke, J., et al. (ed.) 31st IEEE ICDE, pp. 519–530 (2015)
24. Xiao, C., Wang, W., Lin, X., Yu, J.X., Wang, G.: Efficient similarity joins for near-duplicate detection. ACM TODS **36**(3), 15:1–15:41 (2011)
25. Zadeh, R.B., Goel, A.: Dimension independent similarity computation. J. Mach. Learn. Res. **14**(1), 1605–1626 (2013)
26. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search: The Metric Space Approach. Springer, New York (2010)
27. Zhang, D., Yang, G., Hu, Y., Jin, Z., Cai, D., He, X.: A unified approximate nearest neighbor search scheme by combining data structure and hashing. In: Proceedings of the 23rd IJCAI, pp. 681–687. AAAI Press (2013)

Internet of Things and Applications

Immune Approach to the Protection of IoT Devices

Andrzej Chmielewski and Maciej Brzozowski(✉)

Faculty of Computer Science, Białystok University of Technology,
ul. Wiejska 45a, 15-331 Białystok, Poland
{a.chmielewski,m.brzozowski}@pb.edu.pl

Abstract. This paper presents an immune approach for securing different types of devices connected to network. This also applies to the technology, called Internet of things (IoT), which growing rapidly from year to year. It was developed to help people in everyday life, to make our life easier. However, such systems of interrelated computing devices with the ability to transfer data over a network is exposed to various types of attacks. Hacker can take the control over the each device connected to the network. As a result, for example, heating system can be switched on at the summer time, a refrigerator do redundant purchases, etc. To fix this problem, we propose to apply our hybrid immune-based algorithm, called *b-v* model, embedded in a reprogrammable FPGA. It base on negative selection which is suitable to protect a huge amount of devices.

Keywords: Artificial immune system · Anomaly detection · Internet of things · FPGA

1 Introduction

The "Internet of things" (IoT) is the concept of basically connecting to the Internet any device, including not only cellphones and tablets, but also toasters, refrigerators, coffee makers, washing machines, etc. Generally, this applies to almost all physical objects we can think of. Anything that can be connected, will be connected in the near future, with ability of talking to each other. This technology was developed to help people in everyday life, to make our life easier. However, such system of interrelated computing devices with the ability to transfer data over a network is exposed to various types of attacks. Hacker can take the control over the each device connected to the network. As a result, for example, heating system can be switched on at the summer time, a refrigerator do redundant purchases, etc. Moreover, data captured from these devices can then be analyzed by unauthorized persons. Usually such data are sensitive, what tends to loss of anonymity.

It is envisaged, the number of IoT connected devices will number 38.5 billion in 2020, up from 13.4 billion in 2015, which corresponds to rise of over 285 %.

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 75–92, 2016.
DOI: 10.1007/978-3-319-48057-2_5

It means, the security systems has to be ready for protecting the huge amount of hosts, sending a sensitive data, which should not be captured by unauthorized persons.

System of huge amount of connected devices can be compared to *Natural Immune System* (NIS), which efficiency is unsurpassed for all protection systems and verified over millions of years by living organisms. NIS is a very complex system focused on discrimination between own cells (called *self*) and pathogens (called *nonself*), which should be detected and eliminated. A nice feature of NIS is that it does not need any example of *nonself* samples to detect them as only the information about its own cells is sufficient. Hence, every organism has a unique "protection system", capable of detecting even a new type of attack and tolerates only own cells which form its body.

This dedicated and highly efficient protection system against various types of pathogens was an inspiration for developing *Artificial Immune Systems* (AIS). Within this domain, many types of algorithms were proposed, mainly focused on computer system security solutions. However, the most popular is *Negative Selection Algorithm* (*NSA*) [10] with the ability of detecting novel, never met samples, a counterpart of pathogens. Based on deep investigations with various types of large and high-dimensional datasets, a solution called the *b*-*v* model [8] was proposed. It minimizes the problem of scalability, by involving both types of receptors: *b* - and *v*-detectors. This hybrid approach, presented by conducting numerous experiments, provides much better results in comparison to single detection models as well as traditional, statistical approaches, even though only positive (*self*) examples are required at the learning stage. It makes this approach an interesting alternative for well known classification algorithms, like SVM, k-nearest neighbours, etc. The *b*-*v* model is briefly described in Subsect. 3.

To increase the speed of detection process, *b*-*v* model was embedded in a reprogrammable FPGA (Field Programmable Gate Array). Such approach makes, this algorithm can be easily applied to protect also IoT devices.

2 Negative Selection Algorithm

The NSA, i.e. the negative selection algorithm, proposed by Forrest *et al.*, [11], is inspired by the process of thymocytes (i.e. young T-lymphocytes) maturation: only those lymphocytes survive which do not recognize any *self* molecules.

Formally, let $\mathcal{U}$ be a universe, i.e. the set of all possible molecules. The subset $\mathcal{S}$ of $\mathcal{U}$ represents the collection of all *self* molecules and its complement $\mathcal{N}$ in $\mathcal{U}$ represents all *nonself* molecules. Let $\mathfrak{D} \subset \mathcal{U}$ stand for a set of detectors and let $match(d, u)$ be a function (or a procedure) specifying if a detector $d \in \mathfrak{D}$ recognizes the molecule $u \in \mathcal{U}$. Usually, $match(d, u)$ is modelled by a distance metric or a similarity measure, i.e. we say that $match(d, u) = \texttt{true}$ only if $dist(d, u) \leq \delta$, where $dist$ is a distance and δ is a pre-specified threshold. Various matching function are discussed e.g. in [12,15].

The problem relies upon construction the set $\mathfrak{D}$ in such a way that

$$match(d, u) = \begin{cases} \texttt{false} \text{ if } u \in \mathcal{S} \\ \texttt{true} \ \ \text{ if } u \in \mathcal{N} \end{cases} \tag{1}$$

for any detector $d \in \mathfrak{D}$.

A naive solution to this problem, implied by the biological mechanism of negative selection, consists of five steps:

(a) Initialize $\mathfrak{D}$ as empty set, $\mathfrak{D} = \emptyset$.
(b) Generate randomly a detector d.
(c) If $math(d, s) = \texttt{false}$ for all $s \in \mathcal{S}$, add d to the set $\mathfrak{D}$.
(d) Repeat steps (b) and (c) until the sufficient number of detectors will be generated.

Below the binary and real-valued representations of the problem are described.

2.1 Binary Representation

This type of representation was applied by Forrest *et al.* [10] to capture anomalous sequences of system calls in UNIX systems and next to model the system for monitoring TCP SYN packets to detect network traffic anomalies (called LISYS) [13].

In case of binary encoding, the universe $\mathcal{U}$ becomes l-dimensional Hamming space, $\mathbb{H}^l = \{0, 1\}^l$, consisting of all binary strings of fixed length l:

$$\mathbb{H}^l = \{\underbrace{000...000}_{l}, \underbrace{000...001}_{l}, \ldots, \underbrace{111...111}_{l}\}$$

Hence the size of this space is 2^l. The most popular matching rules used in this case are:

(a) r-contiguous bit rule [10], or
(b) r-chunks [2].

Both the rules say that a detector bonds a sample (i.e. data) only when both the strings contain the same substring of length r. To detect a sample in case (a), a window of length r ($1 \le r \le l$) is shifted through censored samples of length l. In case (b) the detector $t_{i,\mathtt{s}}$ is specified by a substring $\mathtt{s}$ of length r and its position i in the string. Below an example of matching a sample by r-detector (left) and r-chunk for affinity threshold $r = 3$ is given

$$\begin{array}{ccc} \overbrace{1\ 0\ \mathbf{0\ 0\ 1}\ 1\ 1\ 0}^{l} & \leftarrow \text{sample} \rightarrow & \overbrace{1\ 0\ \mathbf{0\ 0\ 1}\ 1\ 1\ 0}^{l} \\ 0\ 1\ \underbrace{\mathbf{0\ 0\ 1}}_{r}\ 0\ 0\ 1 \leftarrow & r - \text{detector};\ r - \text{chunk} & \rightarrow *\ *\ \underbrace{\mathbf{0\ 0\ 1}}_{r} *\ *\ * \end{array}$$

Here it was assumed that irrelevant positions in a string of length l representing the r-chunk $t_{3,\mathtt{001}}$ are filled in with the star (*) symbol. This way r-chunk can be identified with schemata used in genetic algorithms: its order equals r and its defining length is $r-1$. Although a single r-detector recognizes much more strings than a single r-chunk, this last type of detector allows more accurate coverage of the $\mathcal{N}$ space [2].

Further, the notion of the ball of recognition allows to define "optimal" repertoire $\mathfrak{D}$. Namely it consists of the detectors located in $\mathbb{H}^l$ in such a way that they cover the space $\mathcal{N}$ and their balls of recognition overlap minimally. A solution to such stated problem was given in [20]. To construct the r-detectors we split all the *self* strings into the templates represented identically as the r-chunks and we construct the detectors by gluing these r-chunks that do not belong to the set $\mathcal{S}$. More formally, if $t_{i,\mathtt{s}}$ and $t_{j,\mathtt{w}}$ are two candidate r-chunks, we can glue them if both the substrings are identical on $r-1$ positions starting from position $i+1$.

Using such an optimality criterion we come to the conclusion that shortest detectors are more desirable as they are able to detect more samples. However, Stibor [18] showed the coherence between r and l values for various cardinalities of S in terms of the probability of generating detectors, P_g. He distinguished three phases:

- Phase 1 (for lower r) – the probability P_g is near to 0,
- Phase 2 (for middle r) – the probability P_g rapidly grows from 0 to 1 (so called *Phase Transition Region*),
- Phase 3 (for higher r) – the probability is very near to 1.

Hence, we should be interested in generating detectors with medium length r (belonging to the second region) and eventually with larger values of r if the coverage of $\mathcal{N}$ is not sufficient. It is worth to emphasize, that the detectors can not be too long, due to exponential increase in the duration of learning process, which should be finished in reasonable time.

2.2 Real-Valued Representation

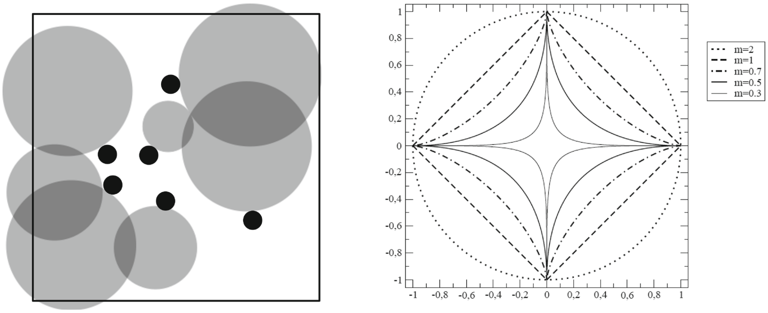
To overcome scaling problems inherent in Hamming space, Ji and Dasgupta [14] proposed a real-valued *NSA*, termed *V-Detector*.

It operates on normalized vectors of real-valued attributes; each vector can be viewed as a point in the d-dimensional unit hypercube, $\mathcal{U} = [0,1]^d$. Each *self* sample, $s_i \in \mathcal{S}$, is represented as a hypersphere $s_i = (c_i, r_s)$, $i = 1, \dots, l$, where l is the number of *self* samples, $c_i \in \mathcal{U}$ is the center of s_i and r_s is its radius. It is assumed that r_s is identical for all s_i's. Each point $u \in \mathcal{U}$ inside any *self* hypersphere s_i is considered as a *self* element.

The detectors d_j are represented as hyperspheres also: $d_j = (c_j, r_j)$, $j = 1, \dots, p$ where p is the number of detectors. In contrast to *self* elements, the radius r_j is not fixed but it is computed as the Euclidean distance from a randomly chosen center c_j to the nearest *self* element (this distance must be greater than r_s, otherwise the detector is not created). Formally, we define r_j as

$$r_j = \min_{1 \leq i \leq l} dist(c_j, c_i) - r_s. \tag{2}$$

The algorithm terminates if a predefined number p_{max} of detectors is generated or the space $\mathcal{U}\backslash\mathcal{S}$ is sufficiently well covered by these detectors; the degree of coverage is measured by the parameter *co* – see [14] for the algorithm and its parameters description.



(a) (b)

Fig. 1. (a) Example of performance *V*-Detector algorithm for 2-dimensional problem. Black and grey circles denotes *self* samples and *v*-detectors, respectively. (b) Unit spheres for selected L_m norms in 2D.

In its original version, the *V*-Detector algorithm employs Euclidean distance to measure proximity between a pair of samples. Therefore, *self* samples and the detectors are hyperspheres (see Fig. 1(a)). Formally, Euclidean distance is a special case of Minkowski norm L_m, where $m \geq 1$, which is defined as:

$$L_m(\mathrm{x}, \mathrm{y}) = \left(\sum_{i=1}^{d} |x_i - y_i|^m \right)^{\frac{1}{m}}, \tag{3}$$

where $\mathrm{x} = (x_1, x_2, \ldots, x_d)$ and $\mathrm{y} = (y_1, y_2, \ldots, y_d)$ are points in $\Re^d$. Particularly, L_2-norm is Euclidean distance, L_1-norm is Manhattan distance, and L_∞ is Tchebyshev distance.

However, Aggarwal *et al.* [1] observed that L_m-norm loses its discrimination abilities when the dimension d and the values of m increase. Thus, for example, Euclidean distance has the best (among L_m-norms) metrics when $d \leq 5$. For higher dimensions, the metrics with lower m (i.e. Manhattan distance) should be used.

Based on this observation, Aggarwal introduced *fractional distance metrics* with $0 < m < 1$, arguing that such a choice is more appropriate for high-dimensional spaces. Experiments, reported in [7], partially confirmed the efficiency of this proposition. For $0.5 < m < 1$, more samples were detected, in comparison to L_1 and L_2 norms. However, for $m < 0.5$ the efficiency rapidly decreased and for $m = 0.2$, none samples were detected. Moreover, these experiments also confirmed a trade-off between efficiency, time complexity and m.

For fractional norms, the algorithm runs slower for lower m values; for $L_{0.5}$ the learning phase was even 2–3 times longer than for L_2.

Another consequence of applying fractional metrics for V-Detector algorithm is modification of the shape of detectors. Figure 1(b) presents the unit spheres for selected L_m-norms in 2D with $m = 2$ (outer most), $1, 0.7, 0.5, 0.3$ (inner most).

3 The *b*-*v* Model

Unsatisfactory coverage of space $\mathcal{N}$ is the main flaw of the v-detectors. To overcome this disadvantage as well as to improve the detection rate (DR for short) and to fasten the classification process, a mixed approach, i.e. b-v model was proposed. Its main idea is depicted in Fig. 2.

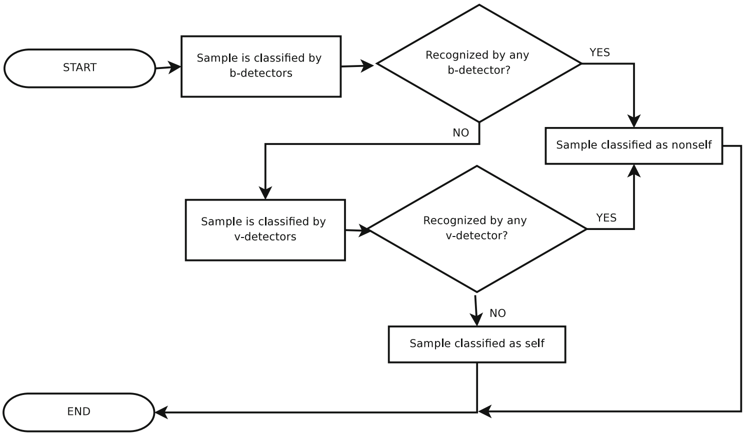

START
Sample is classified by b-detectors
Recognized by any b-detector?
YES
NO
Sample classified as nonself
Sample is classified by v-detectors
Recognized by any v-detector?
YES
NO
Sample classified as self
END

Fig. 2. Flow diagram of the classification process for b-v model.

Here, the b-detectors, as those providing fast detection, are used for preliminary filtering of samples. The samples which did not activate any of the b-detectors are censored by v-detectors next. It is important to note that we do not expect that b-detectors covers the space $\mathcal{N}$ in sufficient degree, as it can consume to much time. More important aspect is their length. They should be relatively short (with high generalization degree) to detect as quickly as possible the significant part of *nonself* samples. The optimal length, r, of b-detectors can be determined by studying the phase transition diagram mentioned in Subsect. 2.1. Namely, we choose the r value guaranteeing reasonable value of the P_g probability what, in addition to ease of generating detectors, results in sufficiently high coverage of the $\mathcal{N}$ space.

In the b-v model the overall DR ratio as well as the average time of detection depends mainly on the number of recognized *nonself* samples by "fast" b-detectors. Thus, in the experiments reported later, we focus mainly on these parameters.

3.1 Building *b*-detectors in Space $\Re^n$

Usually, samples are represented as real-valued vectors. Thus, to construct *b*-detectors, the *self* samples should be converted into binary form first. This can be done in many ways, but probably the simplest one (at least from the computational point of view), is the uniform quantization, [17].

Generally, quantization (used e.g. in digital signal processing, or image processing) refers to the process of approximating a continuous range of values by relatively small set of discrete symbols or integer values. A quantizer can be specified by its input partitions and output levels (called also reproduction points). If the input range is divided into levels of equal spacing, then the quantizer is termed as the uniform quantizer; otherwise it is termed as a non-uniform quantizer, [17].

A uniform quantizer can be described by its lower bound and the number of output levels (or step size). However, in our case, the first of these value is always 0, as we operate only on values from the unit interval (required by the *V*-Detector algorithm). Moreover, for binary representation of output values, instead of the number of output levels, we should rather specify the parameter *bpa*, denoting the number of bits reserved for representing a single level.

The quantization function $Q(x)$ for a scalar real-valued observation x, can be expressed as follows:

$$Q(x) = \lfloor x * 2^{bpa} \rfloor, \tag{4}$$

The resulting integer from the range $\{0, 2^{bpa} - 1\}$ is converted to a bit string of length $l = n * bpa$, where n is the dimension of real-valued samples.

3.2 Representation of *b*-detectors in Space $\Re^n$

V-Detector algorithm can take into consideration already generated *b*-detectors, only if they can be represented in space $\Re^n$. In this case, two different shapes of *b*-detectors in real-valued space were investigated: hyperspheres and hypercubes.

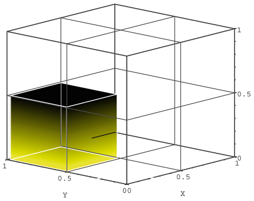


Fig. 3. Representation of *b*-detector `101` in space $\Re^3$ for $bpa = 1$.

The simplest way of converting *b*- to *v*-detector is when $w = r$. Then, the center of *b*-detector (c_{vb}) in space $\Re^w$ can be calculated as follow:

$$c_{vb}[k] = \frac{toInt(b_{k*bpa,(k+1)*bpa-1})}{2^{bpa}} + \frac{1}{2^{bpa+1}}, \quad \text{for } k = 0, \ldots, w-1 \tag{5}$$

where $b_{i,j}$ denotes the substring of b-detector from position i, to j and $toInt$ is the function which returns the decimal value of the binary number. Depending on used shape, the diameter (in case of hyperspheres) or edge (for hypercubes) is equal to 2^{-bpa}. An example of representation of single b-detector in space $\Re^3$ is presented in Fig. 3.

4 Hardware Implementation of the *b*-*v* model

Traditional software firewalls when analysing and filtering packets flowing through the network use the processing power on which they are installed. This can lead to a significant reduction in responsiveness of the computer during a network attack. An alternative to software solutions are hardware firewalls, however, they are relatively expensive to use.

Another solution are firewalls embedded in a reprogrammable FPGA (Field Programmable Gate Array) architectures characterized by a high degree of flexibility during the design process. Usage of HDL (Hardware Description Language) languages reduces the cost of design and allows to transfer design between different architectures from manufacturers such as Xilinx or Altera. The use of reprogrammable architecture for the construction of a firewall reduces the cost of the final solution and increases the number of classified packets compared to pure software solution.

Construction of FPGA allows for parallelization of calculations and algorithms, so that they have wide range of applications during the HPC process (High Performance Computing). Another advantage of FPGAs is their low power consumption compared to the GPU (Graphics Processing Unit) solutions used in HPC, so that they represent better value performance-per-watt power consumption [19].

The use of FPGAs significantly shortens time to market (the length of time since the inception of the product concept to placing it on the market) compared to system based on ASIC (Application Specific Integrated Circuit).

The biggest advantage of FPGAs is that they can be reprogrammed even after they have been installed in the target system. This allows to correct the errors or complement the design with new functionality. In the case of ASIC the process of design and manufacturing should be repeated and then replace a malfunctioning chip in the target system, which is associated with high costs.

The research are focused on building firewall with high throughput and latency as low as possible through the use of reprogrammable architectures. Designed in this way, the firewall will work independently without supervision of other (external) devices.

FPGAs contain programmable logic (configurable logic blocks) contains a set number of LUTs (LookUp-Table - small memory generate boolean function), flip-flops and multiplexers connected via programmable interconnects.

As the implementation language of hardware firewall has been chosen VHDL (Very High Speed Integrated Circuits Hardware Description Language) because is the most supported hardware description language by synthesis software and source code simulators.

4.1 *b*-detectors - Combinational Approach

In initial stage of work on the hardware firewall design was considered as a purely combinational circuit. The advantage of this approach is the clarity of its timing behavior (input/output response) analysis in oder to verify its correctness. Another advantage of this approach is the eases of making changes in the design description.

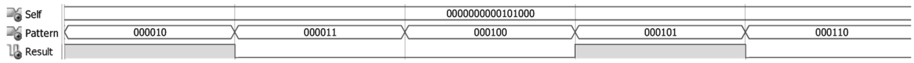


Fig. 4. Combinational component approach: simulation waveform (timing diagram) of anomaly detection.

Figure 4 shows a simulation of generating detectors for combinational component approach. In order to increase the readability of input/output values, results of operations component the were limited to a $l = 16$ for each sample and $r = 6$ for r-chunks. On simulation sample was market as Self - on the input might be inserted signal values of Self/NonSelf; detectors as $Pattern$ - r-chunks; and the $Result$ as the output of designed unit. $Value$ of '1' on output denotes that the detector bonds a sample. For example, if $Self = \{0000000000101000\}$ and $Pattern = \{000010\}$ then $Result = 1$ ($Self$ and $Pattern$ are matched). The same behavior is observed for larger values $l = 64$ and $r = 32$ or $l = 128$ and $r = 64$ etc. The only restriction is length of $1 \leq r \leq l$.

The proposed solution indicates very regular structure. Therefore, a different approach - pipelined - should be considered.

4.2 *b*-detectors - Pipelined Approach

Throughput and delay are two critic performance criteria for designed component. Delay is the time that one task required to be completed, and throughput is the number of tasks that can be completed in one unit time. The major step of adding pipeline to immunologic anomaly detection design into stages. As was mentioned before combinational version indicates very regular structure. Comparison of r-chunk vector witch parts of self/nonself vector is realized on LUT tables which have the same or very close critical propagation delay time. An analogous situation occurs with the logical summation of the results of the comparison which is realized by a cascade connected LUT tables. Therefore when converting the combinational version of immunologic design to pipeline component version the system delay would not change but the system throughput will increase.

Figure 5 shows a simulation censoring samples for pipeline component approach. In order to increase the readability of input/output values results of operations component the were limited to a $l = 23$ for each sample and $r = 8$ for r-chunks. Clk is a signal used for computing stages synchronization. Test signal

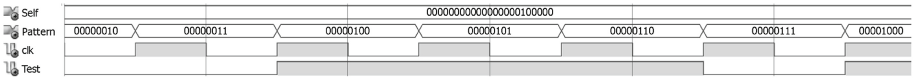


Fig. 5. Pipeline approach: simulation waveform (timing diagram) of anomaly detection.

value is delayed for 4 ticks of the clock. It means that the system needs 4 ticks of the clock to calculate value of Test signal and its directly depend on length of l and r.

Table 1. Comparison of combinational and pipeline versions.

l	r	tc [ns]	tp [ns]	n
11	8	9.101	7.234	3
23	8	11.133	6.869	4
71	8	20.201	7.486	5
35	32	11.801	7.454	4
47	32	18.335	7.215	5
95	32	14.615	7.906	6

4.3 Comparison of Hardware *b*-detector Implementations

Table 1 shows comparison of two approaches, combinational and pipeline, of matching patterns and samples. All measurements were made for a Xilinx xc3s250e-5pq208 device from Spartan3E family. Parameter tc describe maximum combinational path delay for combinational version of matching system, tp - maximum combinational path delay for pipeline version and n how many ticks of the clock is needed to compute output value. It means that for the length $l = 47$ and $r = 32$ combinational system version may match 54 M of sample/pattern pairs and pipeline - 138 M - but the system response take 5 clock ticks (36,075 ns).

4.4 *V*-Detectors

As was already mentioned, classification with the V-Detector algorithm is a very complex and time consuming process because of its high complexity of performed calculations that need to be done on real-valued vectors. Time of classification of a single sample directly depends on its dimensionality ($\Re^d$) and number of detectors, because for each single sample, there is a need to calculate distance to generated detectors from set $\mathfrak{D}$. The described process, of course, might be divided into sub tasks to parallelize classification. As a result, the duration of this process should be significantly decreased. However, when calculations are performed on CPUs or GPUs processors, its efficiency is strictly restricted by number of processors and numbers of its cores.

One of the solutions, corresponding to the above restrictions, are programmable devices, especially Field Programmable Gate Array (FPGA) devices, providing a high performance, low power consumption and relatively low price in comparison to CPU/FPU solutions. FPGA is mainly designed to parallelize computational tasks. Therefore, it might be used in projects where the main indicator is performance. Designers equipped FPGA devices in specialized IP (Intellectual Property) blocks like multipliers, pre-adders and accumulators for increasing the number of computations per second and other more complicated blocks like embedded CPUs, DSP, Ethernet Physical Interfaces, PCI Express, DRAM controllers and many others. Moreover, some FPGA devices allow for the partial reconfiguration (reprogramming) during its operation - helping the system to adopt to rapidly a changing environment.

FPGA devices are characterized by high computing power, flexibility and scalability. They can be adapted as a base for on-line classification systems eg. firewalls and intruder detection systems for home use as well as for enterprise solutions.

As was presented in [4], *b*-detectors were successfully implemented in reprogrammable architectures, especially in FPGA devices. Hence, it is natural to try to do the same for the *V*-Detector algorithm, where we could expect more spectacular results in comparison to a software approach, which was a bottleneck in the *b*-*v* model.

Classification algorithm will be explained in further sections but firstly we must present how the distance between detectors and samples is calculated. The main difference, in comparison to software implementation, is the usage of hypercubes, instead of hyperspheres. Such a shape is more suitable for reprogrammable architecture.

Each *self* sample, $s_i \in \mathcal{S}$, is represented as a hypercube $s_i = (c_i, l_s)$, $i = 1, \ldots, l$, where l is the number of *self* samples, $c_i \in \mathcal{U}$ is the center of s_i and l_s is half of the length of its side (edge). It is assumed, that l_s is identical for all *self* samples. Similar to software approach, each hypercube $u \in \mathcal{U}$ inside any *self* hypercube s_i is considered as a *self* element.

The detectors d_j are represented as hypercubes also: $d_j = (c_j, l_j)$, $j = 1, \ldots, p$ where p is the number of detectors. In contrast to *self* elements, the half length of its side l_j is not fixed, but it is computed as the distance from a randomly chosen center c_j to the nearest *self* element (this distance must be greater than l_s, otherwise detector is not created).

Formally, we define l_j as:

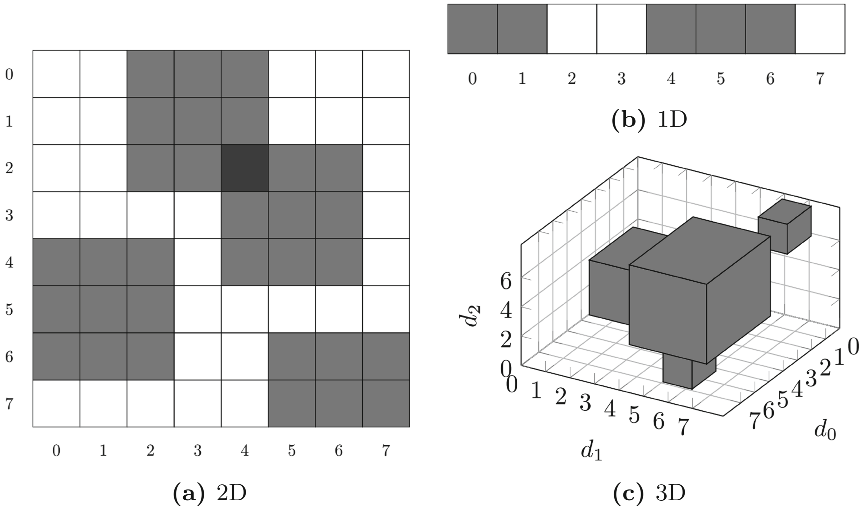
$$l_j = \min_{1 \le i \le l} dist(c_j, c_i) - l_s, \tag{6}$$

where distance between two centers of hypercubes x and y is defined as:

$$dist(x, y) = \min_{1 \le i \le d} |x_i - y_i|. \tag{7}$$

Each hypercube is axis-aligned. It means, rotation of hypercube is not allowed. Two hypercubes are not overlapping when:

$$\min_{1 \le i \le d} |x_i - y_i| > l_x + l_y. \tag{8}$$



(b) 1D

(a) 2D

(c) 3D

Fig. 6. Example of axis-aligned v-detector hypercube in $\Re^d$.

Figure 6 presents d-dimensional hypercubes. In Fig. 6(b) two detectors $d_1 = ([0], 1)$, $d_2 = ([5], 1)$ in one dimensional space are marked (grey colour). In Fig. 6(a) two detectors $d_1 = ([1, 3], 1)$ and $d_2 = ([3, 5], 1)$ are overlapping. In Fig. 6(c): $d_1 = ([1, 1, 1], 0)$, $d_2 = ([0, 6, 7], 0)$, $d_3 = ([5, 5, 0], 0)$ and $d_4 = ([5, 5, 5], 1)$ are examples in three dimensional space. In V-Detector classification process it is sufficient to state that sample overlap with one of the detectors. There is no need to state that detector includes the sample (sample overlaps a whole over the detector or overlaps its partially). In both cases, the sample should be rejected.

4.5 Emebedded V-Detector- the Fastest Approach

The first of the proposed solutions, denoted as V-Detector$-HFast$, for reprogrammable architectures holds the entire $\mathfrak{D}$ in target device resources. As a result, it should maximize the speed of classification process for V-Detector algorithm. The base concept allows the determination of distance between the single sample and the entire set of $\mathfrak{D}$ in one cycle of the designed system. Moreover, the classification process does not depend to such an extent on the number of dimensions as in the classical approach. Here, the universe set $\mathcal{U}$ is represented

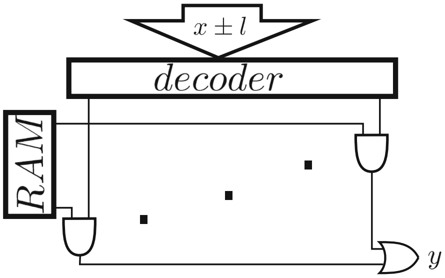


Fig. 7. V-Detector- the fastest approach for reprogrammable architectures.

by bitmap with detectors modeled as hypercubes. The number of distinguish samples (represented as vectors) depends on the size of used internal memory.

Memory and other arithmetic operations are mapped into logic elements available on the reprogrammable device, in our case FPGA. For each dimension, a decoding vector responsible for choosing data from set $\mathfrak{D}$ is created, depending on the sample's coordinates. For each coordinate logical products are created of decoding vectors (for all dimensions) and its corresponding memory values. In the last step, the number of '1' in the logical product is counted. If this value is not equal to 0, it means, sample overlaps at least one detector. Figure 7 shows a simplified diagram of V-Detector(for one dimension) based on internal FPGA memory.

The proposed solutions for reprogrammable architectures is incredibly fast in classification of both single and group of samples. This approach, apart from the high-speed classification, also has some disadvantages. Set $\mathfrak{D}$ is held in FPGA internal RAM (very limited capacity) or is mapped in the logical area on the device. In this case, a synthesized classification algorithm is highly demanded for device logic resources. Therefore, we are limited by logic element numbers on the target programmable device, in which one, design has to be mapped. In this case we have to choose a target device with appropriate logic elements overhead. The lack of adequate number of logical resources may lead to a decrease in the number of analyzed dimensions, or enforces the reduction of the resolution of the analyzed samples.

4.6 Embedded V-Detector - Approach Based on External Memory

The presented classification solution, denoted as V-Detector$-HRAM$, is based only on the resources provided by the target device, is characterized by constraints on the system resolution. One of the methods to increase the resolution of the system is to move the set of detectors from the internal device resources to external memory. In this case, part of the available resources has to be destined to calculate the read/write memory cell addresses. The designed component informs which memory lines are needed in the calcification process of each sample and chooses only the appropriate cells from them. The proposed solution is slowed down by time needed by external memory to perform write and read operations compared to the approach presented in Subsect. 4.5. When the system is properly scaled to expected samples (in numbers of dimensions and numbers of bits per dimension) i.e. $l_s = 0$ (is small as possible) classification process should need only one read operation from external memory.

For DE2-115 Development Board we were able to achieve a resolution of 2^{30} distinguishable samples.

5 Hardware *V*-Detector Approaches Evaluation

In the proposed solutions, the learning phase is simplified, in comparison to software implementation. The optimization process of $\mathfrak{D}$ set is not needed because its size does not affect the duration of classification of a single sample. Bitmap hypercube representation of $\mathcal{U}$ is constant and therefore the number of detectors has no impact on its size. Classification time is shorter because the designed components consider calculations only in the nearest surroundings of the sample. All the necessary calculations are performed at the level of the FPGA. Therefore, the designed component may process large amounts of data in a single period of time.

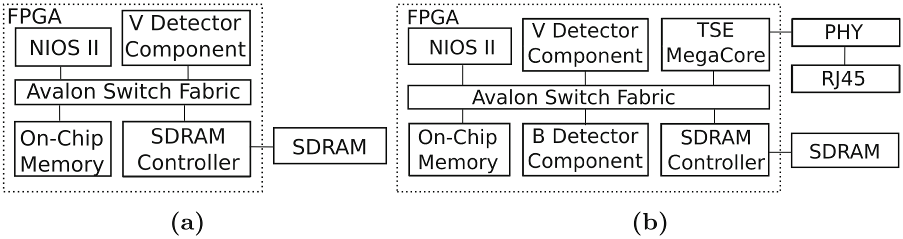


Fig. 8. System diagrams for FPGA device.

Both presented solutions were implemented and synthesized for Altera Cyclone IVE EP4CE115F29C7 device equipped on the DE2-115 Development and Education Board by using Quartus II 15.0. Figure 8(a) shows simplified diagrams of the actual designs. The core of the solution is Nios II processor with connected components via an Avalon bus. In the first stage, components were tested as operated independently. In the second stage, components were integrated into the system (Fig. 8(a)). In the future, both systems, *V*-Detector based on internal and on external RAM cells, will be equipped with two Ethernet interfaces 10/100/1000 and tested in a network environment (Fig. 8(b)). Another possibility of the further development is integrated design with PCI Express and use also of the desktop computer resources.

To measure performance (profiling) of the designed systems we used Altera Megacore Performance Counter Unit. The designed system was based on Nios II worked with 100 MHz frequency.

As was already mentioned in Subsect. 4.5, the dimensionality of samples which can be processed by V-Detector$-HFast$ algorithm is highly restricted by resources availability in target devices. In the case of Altera Cyclone IVE EP4CE115F29C7, which was used in our experiments, internal memory capacity is limited to 3888 KB. Such a restriction forced us to select a rather small dataset for testing. We chose the most popular dataset used for classification purposes, namely $Iris$, which consists of only 150 samples. Each of 4 attributes was represented by integer value from the range 0–3 (2 bits). Detectors were generated, assuming that one class of thew iris plant is regarded as set $\mathcal{S}$ during the learning stage. In all cases, samples were successfully classified. The main advantage of the hardware approaches was the duration of classification, and was presented in Table 2.

Table 2. Comparison of average duration of classification process for V-Detector$-HFast$ and V-Detector$-HRAM$ algorithms for different number of bits per dimension (n) for $Iris$ dataset.

	$HFast$		$HRAM$			
	n=2		n=2		n=6	
	time[ms]	time (clock)	time[ms]	time (clock)	time[ms]	time (clock)
Iris-setosa	0.18	17914	0.58	58181	0.58	58181
Iris-versicolor	0.18	17914	0.58	58061	0.58	58061
Iris-virginica	0.18	17914	0.58	58038	0.58	58038

We can compare those values with the average time of classification achieved for software implementation executed on a PC equipped with Intel i7-3770 processor (8 cores) 3.4 GHz with 16 GB RAM. In this case, process censoring all samples took about 3 ms. It is about 20 times slower than V-Detector$-HFast$, and about 6 times slower than V-Detector$-HRAM$. Taking into consideration that the frequency of the CPU on the PC computer was 340 times higher, it can be easily computed how fast the hardware approach is.

Table 2 contains also the results of other classification experiments with the same dataset. Here, samples were represented as 4 dimension vectors with 6 bits per dimension. For V-Detector$-HFast$ this design was too big to map into available architecture. Thus experiments were conducted only for the V-Detector$-HRAM$ algorithm for which the classification process took 0.58 ms (the same time as in first experiment). Here, classification strictly depends on access time to external memory (read/write data operation). It is clear that all operations performed on internal resources are faster than on external memory. The increase in the number of dimensions does not significantly affect the time of classification. The most important is a properly scaled system (sample size).

However, the most spectacular results were obtained for some subsets of KDD Cup 1999 from UCI Machine Learning Repository. This database was too big for the current implementation of V-Detector$-HRAM$. Hence, only the samples of the ICMP protocol were used for tests with randomly selected 7 of 41 attributes

Table 3. Comparison of average duration of classification process for software implementation and V-Detector$-HRAM$ algorithm applied to KDD Cup 1999 subset.

$Software$	$HRAM$	
time[ms]	time[ms]	time (clock)
106 000	45	4483698

(each coded on 4 bits). In this way, our test set contains 1074994 unique samples and an average of 683 detectors were used. During the tests we could observe, the duration of classification was more than 2000 times faster in the case of hardware implementation (see Table 3).

Table 4. Comparison of average duration of classification process for software implementation and V-Detector$-HRAM$ algorithm applied to KDD Cup 1999 TCP and UDP subset.

	$Software$	$HRAM$	
	time[ms]	time[ms]	time (clock)
UDP 2000 samples (RAM only)	not tested	2.24	224002
UDP 84545 samples (SD Card)	23000	1578	157817722
TCP 978539 samples (SD Card)	617000	18226	1822604218

Capacity of our system was to small to hold samples for tests of UDP and TCP subset. For UDP was tested 84545 samples and for TCP 978539 samples. Therefore there was a need to store data in portable memory like SD cards. During the tests we could observe, the duration of classification was more than 14 times faster in the case of hardware implementation for UDP set and more than 30 time for TCP subset (see Table 4). The slowdown in the classification process results from time needed to access external memory. Usage of external memory (SD Card) slowing down the classification process more than 60 times.

6 Conclusions

One major disadvantage of algorithms based on negative selection is scalability. The use of two types of detectors have a positive impact mainly on the efficiency of the detection process. However, only the hardware implementation of the b-v model in FPGA significantly reduced detection time and opened new possibilities for applications.

Conducted experiments confirmed the effectiveness of the model and its suitability for the protection of various types of embedded systems, including IoT devices. Some improvements are still necessary and relate primarily to changes proposed solutions for working with large datasets with a significant number of dimensions. In its present form, our solution should rather be considered as a support system for existing solutions in order to detect new types of attacks.

Acknowledgment. This research was partially supported by the grants S/WI/3/13 and MB/WI/1/2014 of the Polish Ministry of Science and Higher Education.

References

1. Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the surprising behavior of distance metrics in high dimensional space. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 420–434. Springer, Heidelberg (2000). doi:10.1007/3-540-44503-X_27
2. Balthrop, J., Esponda, F., Forrest, S., Glickman, M.: Coverage and generalization in an artificial immune system. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002), New York, pp. 3–10, 9–13 July 2002
3. Brzozowski, M., Chmielewski, A.: Embedding the *V*-detector algorithm in FPGA. In: Saeed, K., Homenda, W. (eds.) CISIM 2016. LNCS, vol. 9842, pp. 43–54. Springer, Heidelberg (2016). doi:10.1007/978-3-319-45378-1_5
4. Brzozowski, M., Chmielewski, A.: Hardware approach for generating *b*-detectors by immune-based algorithms. In: Saeed, K., Snášel, V. (eds.) CISIM 2014. LNCS, vol. 8838, pp. 615–623. Springer, Heidelberg (2014). doi:10.1007/978-3-662-45237-0_56
5. Chu, P.P.: RTL Hardware Design Using VHDL: Coding for Efficiency, Portability, and Scalability. Wiley-Interscience, Hoboken (2006)
6. de Castro, L., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer, London (2002)
7. Chmielewski, A., Wierzchoń, S.T.: On the distance norms for multidimensional dataset in the case of real-valued negative selection application. Zeszyty Naukowe Politechniki Białostockiej, No. 2, pp. 39–50 (2007)
8. Chmielewski, A., Wierzchoń, S.T.: Hybrid negative selection approach for anomaly detection. In: Cortesi, A., Chaki, N., Saeed, K., Wierzchoń, S. (eds.) CISIM 2012. LNCS, vol. 7564, pp. 242–253. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33260-9_21
9. Dasgupta, D., Forrest, S.: Novelty detection in time series data using ideas from immunology. In: Fifth International Conference on Intelligent Systems, Reno, Nevada, 19–21 June 1996
10. Forrest, S., Hofmeyr, S. A., Somayaji, A., Longstaff, T. A.: A sense of self for Unix processes. In: Proceedings of the 1996 IEEE Symposium on Research in Security and Privacy, pp. 120–128. IEEE Computer Society Press (1996)
11. Forrest, S., Perelson, A., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of the IEEE Symposium on Research in Security and Privacy, Los Alamitos, pp. 202–212 (1994)
12. Harmer, P.K., Wiliams, P.D., Gunsch, G.H., Lamont, G.B.: Artificial immune system architecture for computer security applications. IEEE Trans. Evol. Comput. **6**, 252–280 (2002)
13. Hofmeyr, S., Forrest, S.: Architecture for an artificial immune system. Evol. Comput. J. **8**(4), 443–473 (2000)
14. Ji, Z., Dasgupta, D.: Real-valued negative selection algorithm with variable-sized detectors. In: Deb, K., Tari, Z. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 287–298. Springer, Heidelberg (2004). doi:10.1007/978-3-540-24854-5_30

15. Ji, Z., Dasgupta, D.: Revisiting negative selection algorithms. Evol. Comput. **15**(2), 223–251 (2007)
16. Keogh, E.J., Chu, S., Hart, D., Pazzani, M.: Segmenting time series: a survey and novel approach. In: Last, M., Kandel, A., Bunke, H. (eds.) Data Mining in Time Series Databases, pp. 1–22. World Scientific, Singapore (2004)
17. Sayood, K.: Introduction to Data Compression. Elsevier, Amsterdam (2005)
18. Stibor, T.: Phase transition and the computational complexity of generating r-contiguous detectors. In: de Castro, L.N., Von Zuben, F.J., Knidel, H. (eds.) ICARIS 2007. LNCS, vol. 4628, pp. 142–155. Springer, Heidelberg (2007). doi:10.1007/978-3-540-73922-7_13
19. Vanderbauwhede, W., Benkrid, K.: High-Performance Computing Using FPGAs. Springer, New York (2013)
20. Wierzchoń, S.T.: Generating optimal repertoire of antibody strings in an artificial immune system. In: Kłopotek, M.A., Michalewicz, M., Wierzchoń, S.T. (eds.) Intelligent Information Systems. Advances in Soft Computing, vol. 4, pp. 119–133. Springer, Heidelberg (2000)
21. Wierzchoń, S.T.:Deriving concise description of non-self patterns in an artificial immune system. In: Jain, L.C., Kacprzyk, J. (eds.) New Learning Paradigm in Soft Comptuning, pp. 438-458. Physica-Verlag (2001)

Implementation and Performance Evaluation of Vehicle-Moving Based Routing Protocol in VANET

Vo Que Son[1](✉), Ta Tri Nghia[1], Nguyen Ho Ba Hai[2], and Nguyen Binh Phuong[3]

[1] Telecommunications Department, University of Technology, Ho Chi Minh City, Vietnam
{sonvq,nghiatt}@hcmut.edu.vn
[2] HITACHI Ltd., Ho Chi Minh City, Vietnam
513460526@hcmut.edu.vn
[3] Office for International Study Programs, University of Technology, Ho Chi Minh City, Vietnam
phuongnb@oisp.hcmut.edu.vn

Abstract. Vehicular Ad-hoc Networks (VANET) has been becoming a potential candidate to enable variety of applications supporting traffic safety, traffic efficiency, and infotainment. There have been numerous research works to address many challenges in VANET to make this technology become possible deployments. One of key issues is to design a scalable routing protocol, which can support the robustness of the route disruption caused by the vehicle mobility. In this paper, a design and implementation of Vehicle-Moving based Routing Protocol is proposed to utilize vehicles' direction information to target the prediction of possible link-breakage events before they occur. This proposed scheme also helps reduce the routing information exchanged by choosing neighbor nodes which are grouped according to the same velocity vector. Besides, performance of the routing scheme is simulated and compared with the other popular protocols to illustrate advantages of the proposed routing strategy in terms of throughput, delay, and packet loss.

Keywords: VANET · Routing · ITS applications · Performance evaluation

1 Introduction

Today, there is a huge number of vehicles joining the traffic system to increase the research interest in the field of vehicle communication [1]. In this direction, several wireless vehicle communication techniques have been studied to target the traffic efficiency, which lay a foundation for Intelligent Transport Systems (ITS) [2–5]. The key mission of ITS is to provide the seamless connections for mobile vehicles to support the safety [6, 7] or to provide traffic conditions with low cost [8].

VANET [9] is a special kind of Mobile Ad-hoc Networks (MANET) in terms of dynamic topology due to the high-speed movement of vehicles (e.g. cars or trucks). However, the mobility in VANET is constraint to a given path of the vehicles and the moving velocity is restricted by speed limit or the traffic congestion conditions on streets. In addition, vehicles can be equipped with RF long-distance transceivers, high power

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 93–104, 2016.
DOI: 10.1007/978-3-319-48057-2_6

supply, and large capacity of data storage. Hence, the data processing and storage capability in VANET is not the main challenges as those in MANET. In addition, the communication of vehicles in VANET can be V2V (vehicle-to-vehicle or V2I (vehicle-to-infrastructure or vehicle to RSU).

In order to bring VANET into wide deployment [8], beside the wireless access techniques [10–13], one of the main issues is to design routing algorithms which support scalability, adaptation to the disrupted radio links between vehicles during their movements. Although there are several similarities between VANET and MANET, the routing protocols designed for MANET are usually not suitable for VANET due to some special characteristics of VANET such as movement information, vehicle positions, or digital map of roads.

Current efforts for vehicle communication is to optimize the routing protocols used in MANET to meet requirements of VANET. These routing schemes can be basically categorized into three types: reactive, proactive and hybrid schemes. In Reactive Routing Protocols (RRPs), the route determination is based on the data transmission of the sending node. If a node has available data to transmit, it will flood the Route Request (RREP) to reach the destination node. When it receives the Route Reply (RREQ), it will cache this route and send data packets to the destination node by using this route. Every link breakage will be notified to all involved nodes by a Route Error message (RERR) for rediscovery of the new route when necessary. The well-known candidates falling in this routing schemes are AODV (Ad-hoc on Demand Distance Vector) [14] and DSR (Dynamic Source Routing) [15]. The big disadvantage of the reactive routing scheme is slow adaption to the fast network topology change due to high speed movement of vehicles in VANET. In addition, the high delay of route discovery mechanism cannot fit in some time-sensitive ITS applications such as CCA (Cooperative Collision Avoidance).

Different from reactive routing, all Proactive Routing Protocols (PRPs) exchange the routing information periodically. This mechanism allows determining any route to any node inside the network at any time. However, the bandwidth consumed by proactive routing is usually higher than that in reactive routing to meet the high adaption of the frequent network topology changes. OLSR (Optimized Link State Routing) [16], HSLSR (Hazy Sighted Link State Routing) [17], TBRPF (Topology Broadcast based on Reverse Path Forwarding) [18], and DSDV (Destination- Sequenced Distance Vector) [19, 20] are the popular proactive protocols falling in this category.

Hybrid Routing Protocols (HRPs) tries to combine the advantages of both reactive and proactive schemes. In this kind of routing, ZRP (Zone Routing Protocol) [21] is an interesting protocol which divides the network topology into many different zones. The routing inside zones (intra-zone routing) is performed by a proactive routing protocol to reduce the transmission delay between nodes inside one zone. Alternatively, in order to increase the network scalability, the inter-zone communication is connected by using a reactive routing protocol.

Based on the routing concepts mentioned previously, several routing protocols are proposed for vehicles' communication in VANET such as CarNet [22] and MOPR (Movement Prediction Based Routing) [23]. CarNet is an application for a large ad-hoc vehicle network to transmit packets without the need of network infrastructure. It places radio vehicles inside grids [24] and utilizes the geography information of vehicles to relay the packets without flooding the network. CarNet supports IP connections and applications related to traffic congestion monitored on high ways or vehicle tracking. Another approach used in MOPR is to predict the future positions of vehicles so that one node can choose the best route to the destination based on the selection of intermediate nodes with most stable links. The quality of link stability between nodes during the movements is estimated from the vehicles' direction.

With all the issues discussed previously, in this paper a Vehicle-Moving based Routing Protocol (VMRP) is proposed to meet several following requirements for a routing protocol used in VANET support V2V and V2I communications:

- Utilizing the proactive routing advantages to support the frequent and dynamic network topology
- Combining the vehicles' information such as position, velocity to estimate the link stability between vehicles.
- Supporting loop free, fast convergence and low complexity for implementation.
- Supporting neighbor discovery with low latency.
- Supporting multi-cast communication.

This paper is structured in 4 sections: Sect. 1 is the introduction to the scope of this paper. Section 2 describes the proposed routing protocol. Simulation results are shown and discussed in Sect. 3. Finally, conclusions and outlook are given at the end of this paper.

2 Design of Vehicle-Moving Based Routing Protocol

2.1 Radio Link Disruption and Vehicle Grouping Mechanism Based on Moving Direction

Related to the link disruption between vehicles, a typical scenario shown in Fig. 1. There are 5 moving cars performing wireless communication among them. The data transmission from the source node A to the destination node E can be successfully carried out using 2 paths: A-C-D-E or A-B-D-E. However, at the cross road, the vehicle B turns

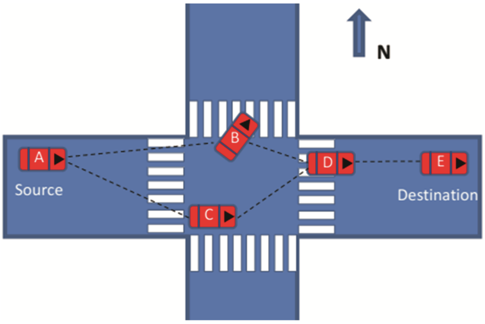


Fig. 1. The possibility of radio link disruption can happen among vehicles due to movements.

left to a new road and the 4 remaining cars continues moving on the old road. This event can lead to the problem that the link between vehicle A-B and B-D can be disrupted causing the end-to-end communication between A and E is unsuccessful.

In order to solve this problem, a mechanism of grouping vehicles moving in the same direction is proposed as follows:

- All vehicles are categorized in 4 groups based on their velocity vectors as shown in Fig. 2. In the Cartesian coordinate each group is indicated by a unit vector $S_1 = (1, 0)$, $S_2 = (0, 1)$, $S_3 = (-1, 0)$, $S_4 = (0, -1)$.

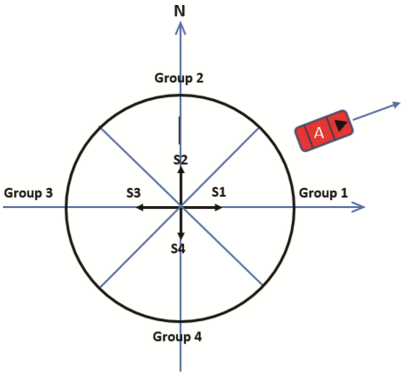


Fig. 2. Vehicle grouping based on velocity vectors.

- All vehicles are assumed to be equipped with GPS devices to determine their geographical positions periodically and convert these position parameters to Cartesian coordinates.
- If the velocity vector of a vehicle illustrated in Cartesian coordinate is (V_x, V_y), performing the multiplication the velocity with 4 unit vectors, the vehicle belongs to the group which has this maximum multiplication result.

Assuming the current position of a vehicle is (X_1, Y_1) and the previous one is (X_0, Y_0) (shown in Fig. 3), the velocity vector (or direction vector in Δt) $V_A = (V_x, V_y)$ can be calculated by using the following equations:

$$\begin{cases} V_x = \left(X_1 - X_0\right) / \Delta t = X_1 - X_0 \\ V_y = \left(Y_1 - Y_0\right) / \Delta t = Y_1 - Y_0 \end{cases} \quad (1)$$

In order to determine the group that the vehicle having V_A belongs to, the multiplication of V_A with unit direction vectors of groups is performed as follows:

$$\begin{cases} V_A.S_1 = \left(V_X, V_Y\right).(1, 0) = 1.V_X + 0.V_Y \\ V_A.S_2 = \left(V_X, V_Y\right).(0, 1) = 0.V_X + 1.V_Y \\ V_A.S_3 = \left(V_X, V_Y\right).(-1, 0) = (-1).V_X + 0.V_Y \\ V_A.S_4 = \left(V_X, V_Y\right).(0, -1) = 0.V_X + (-1).V_Y \end{cases} \quad (2)$$

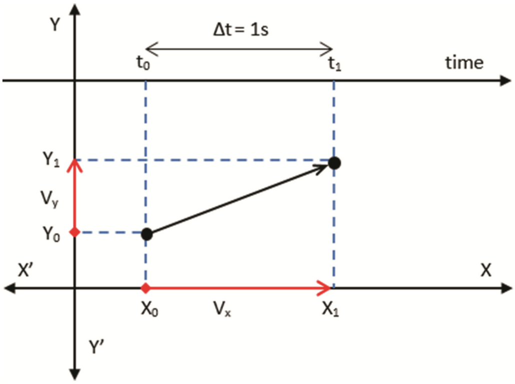


Fig. 3. Determining the velocity vector (V_x, V_y) from 2 continuous positions with the period $\Delta t = 1$ s.

For example in Fig. 2, the result ($V_A.S_1$) will give the maximum value using (2); hence, the vehicle A belongs to group 1 in this figure.

2.2 Vehicle-Moving Based Routing Algorithm

In ad-hoc networks using proactive routing protocols, the routing information contained in *Hello Messages* (or *Route Update/Control Message*) is periodically exchanged among nodes. The common format of this message displayed in Fig. 4 consists of several important fields such as Next Hop, Metric & Sequence for each destination. The group information (calculated in the previous section) is proposed to be integrated in this routing message. When a vehicle receives a *Hello Message* from other vehicle, it will compare its *Group ID* with the *Group ID* of the sending vehicle. If the sending vehicle and the receiving vehicle belongs to different groups, the link between them is considered unstable. Then, a specific group metric will be added into the routing metric between 2

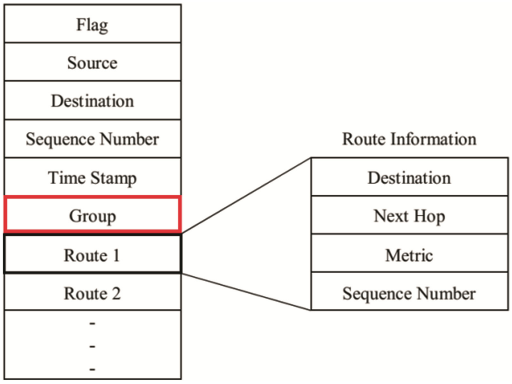


Fig. 4. Typical format of *Hello Message* with integrated group information.

vehicles and the updated routes. Based on this mechanism, the group metric will reflect the group information via the routing protocol used by vehicles.

Otherwise, routing metrics are not changed and the routing algorithm is performed based on the number of hops like the basic Bellman-Ford.

In Fig. 1, β_{AB}, β_{BD}, β_{AC}, β_{CD} are used to denote the routing metrics of the radio link between vehicles A and B, B and D, A and C, C and D. In case of no group metric, all these routing metrics are set equal to 1 and both routes A-B-D and A-C-D can be used for the communication between A and E.

However, if a group metric α_m is added to the routing metric β_{AB} and β_{BD} ($\beta_{AB} = \beta_{BD} = 1 + \alpha_m$), the total cost of these two routes will be changed because vehicle B belongs to the group that is different from the group of A and D.

The cost of route A-B-D is given below:

$$\beta_{AB} + \beta_{BD} = 2\left(1 + \alpha_m\right) = 2 + 2\alpha_m \tag{3}$$

Meanwhile the cost of route A-C-D is unchanged $\beta_{AC} + \beta_{CD} = (1 + 1) = 2$. Therefore, the route A-C-D will be selected for data transmission. The proposed mechanism will ensure the stability of routes for communication.

Figure 5 illustrates the algorithm of the grouping process at each node of VANET, in which each vehicle can get the direction information from equipped positioning device (e.g. GPS) in kind of an angle between the moving direction and a standardized direction. In this paper, the East direction is used for normalization. From that information, the velocity vector of a vehicle can be determined easily for the grouping algorithm. Besides, each vehicle also sets the Group ID of the RSU (Road Side Unit) equal to 0 in order to distinguish with the Group ID of itself.

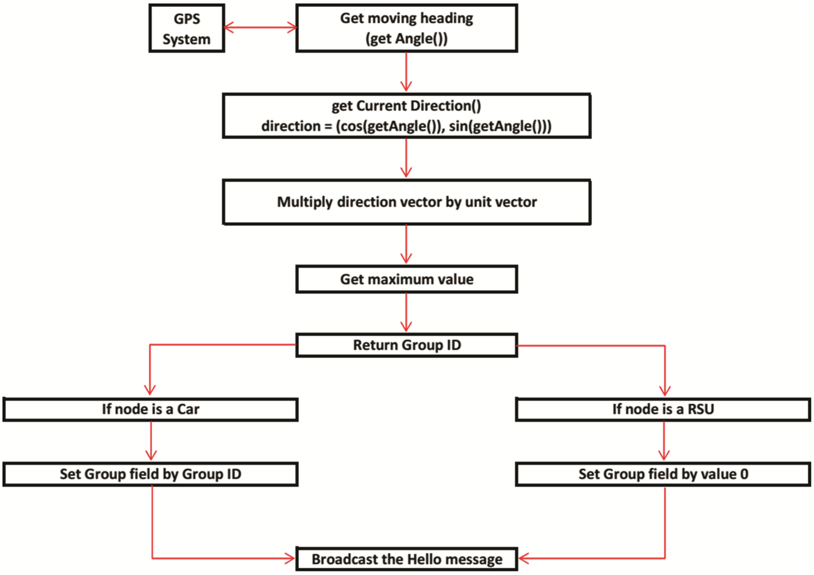

GPS System
Get moving heading (get Angle())
get Current Direction()
direction = (cos(getAngle()), sin(getAngle()))
Multiply direction vector by unit vector
Get maximum value
Return Group ID
If node is a Car
If node is a RSU
Set Group field by Group ID
Set Group field by value 0
Broadcast the Hello message

Fig. 5. Algorithm of vehicle grouping at each node.

The algorithm of processing *Hello Message* is illustrated in Fig. 6. When a vehicle X receives from a vehicle Y this routing message containing the routing metric and a sequence to a destination vehicle A, it will check whether it has any route to the destination A. If there is no such route, the vehicle X will add this route to its routing table; otherwise (if this route exists), it will update the corresponding route entry if one of the following conditions happen:

- The current sequence is greater than that of the old route in the routing database
- The current sequence is equal to the old one but the current routing metric is smaller.

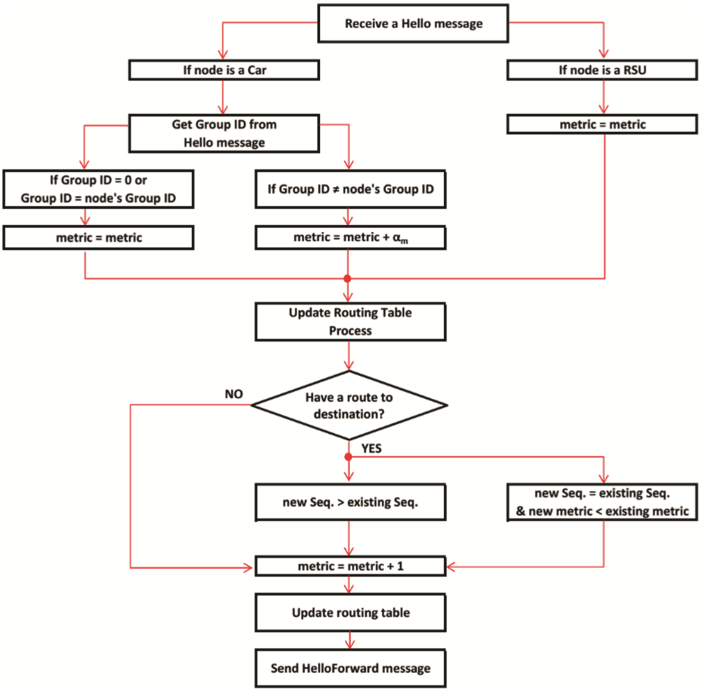

Receive a Hello message
If node is a Car
If node is a RSU
Get Group ID from Hello message
metric = metric
If Group ID = 0 or Group ID = node's Group ID
If Group ID ≠ node's Group ID
metric = metric
metric = metric + α_m
Update Routing Table Process
NO
Have a route to destination?
YES
new Seq. > existing Seq.
new Seq. = existing Seq. & new metric < existing metric
metric = metric + 1
Update routing table
Send HelloForward message

Fig. 6. Algorithm of *Hello Message* processing at each node.

This mechanism works like DSDV routing protocol to eliminate the loop problem in routing. Besides, if the received *Hello Message* has the *Group ID* equal to 0, the message is sent by a RSU. Then, the vehicle does not update the routing metric. Similarly, if a RSU gets a *Hello Message* from any vehicle, it will update the routing table without changing the routing metric despite the group that the sending vehicle belongs.

3 Simulation Results

3.1 Description of Simulated Network

The simulated VANET is set up based on the digital map of District 1, Ho Chi Minh City, Vietnam (shown in Fig. 7) where there are many vehicles joining the traffic system during the day. OmNet++ [25, 26], VEINS [27], and SUMO [28, 29] are combined to simulate the above VANET to achieve the accuracy. In the simulated scenario, there are 6 vehicle flows (with 20–30 vehicles/flow) moving along roads with different directions. The vehicles' speed can be changed from 5 m/s to 15 m/s.

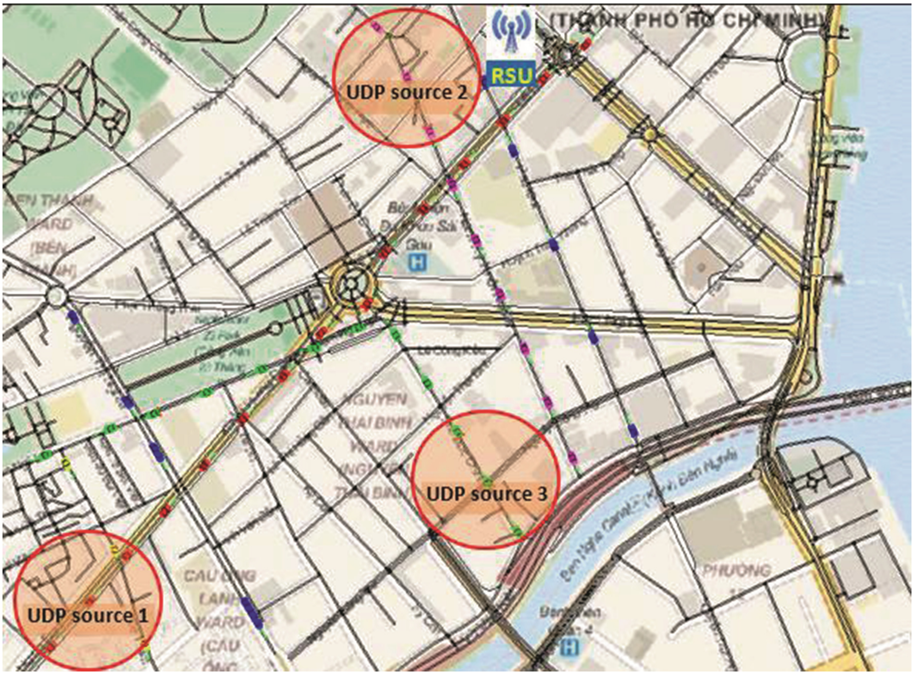


Fig. 7. Simulated VANET in Ho Chi Minh City with flows of vehicles.

In the simulated scenario there is a RSU used for communication with vehicles. Three vehicles use UDP connections to send packets to the RSU via multi-hop networking. The packet size is set at 100 bytes and the date rate of each connection is 12 Mbps. The interval of *Hello Message* exchanged by vehicles is configured at several values of 1 s, 3 s, 5 s, or 10 s to investigate the performance of VMRP. Several parameters are investigated such as end-to-end delay, packet loss ratio, or effect of vehicle speed to those parameters.

In the following section, the performance evaluation of VMRP will be compared with that of DSDV due to the similarity of both these routing protocols.

3.2 Performance Evaluation

The results displayed in Fig. 8 illustrate the packet loss ratio of both DSDV and VMRP with different *Hello Message* intervals. The packet loss ratio is reduced approximately 15.2 % (at *Hello Message* interval of 1 s) and 27.2 % (at interval of 10 s) in comparison with those of DSDV.

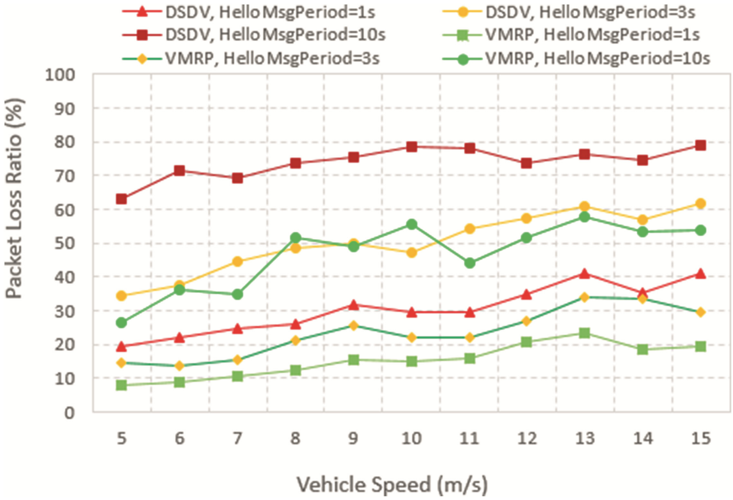


Fig. 8. Packet loss ratio versus vehicle speeds of VMRP and DSDV.

However, there is a relationship between vehicles' speed and the *Hello Message* interval that can affect to the packet loss. When vehicles increase their velocity speeds, the packet loss also increases correspondingly because vehicles cannot predict links' stability precisely. Hence, in order to keep the packet loss rather low (e.g. 15 %) the *Hello Message* period must be kept low enough; for example, 1 s in this simulation.

Considering the end-to-end delay, the result shown in Fig. 9 indicates that there is no big delay difference between DSDS and VMRP at different velocities of vehicles. However, when increasing the *Hello Message* period, the routing traffic in the network also increases due to the exchange of routing messages. This leads to the higher end-to-end delay shown in Fig. 9. In addition, the velocity does not affect significantly to the end-to-end latency.

Therefore, there is an interesting trade-off between the results in Figs. 8 and 9. In order to adapt to the fast movement of vehicles, the lower *Hello Message* interval should be used, which causes higher end-to-end delay and vice versus.

In order to investigate the effect of data rates and vehicles' velocities to the packet loss, the data rates of source nodes are changed to 6 Mbps, 12 Mbps and 24 Mbps at several velocities for simulation. The period of *Hello Message* is set at 1 s for high adaption to the network changes. The results are shown in Fig. 10 and it can be seen that the packet loss ratio of VMRP outperforms that of DSDV (approximately 15 %) despite of data rates and speeds.

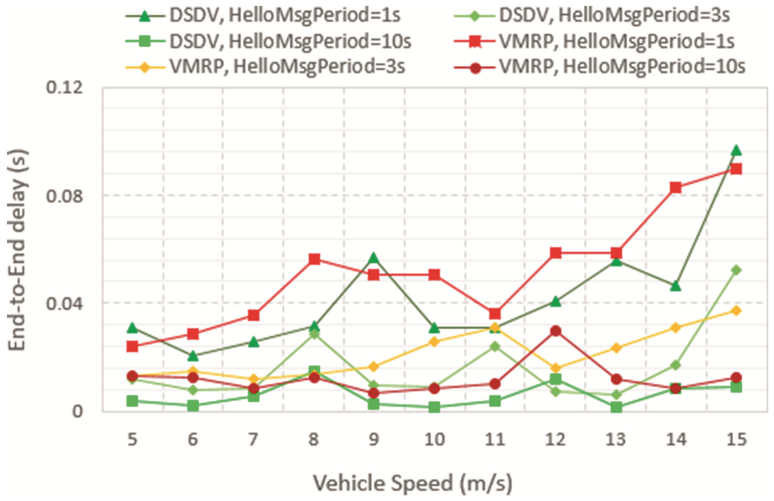


Fig. 9. End-to-end delay versus vehicle speed of VMRP and DSDV.

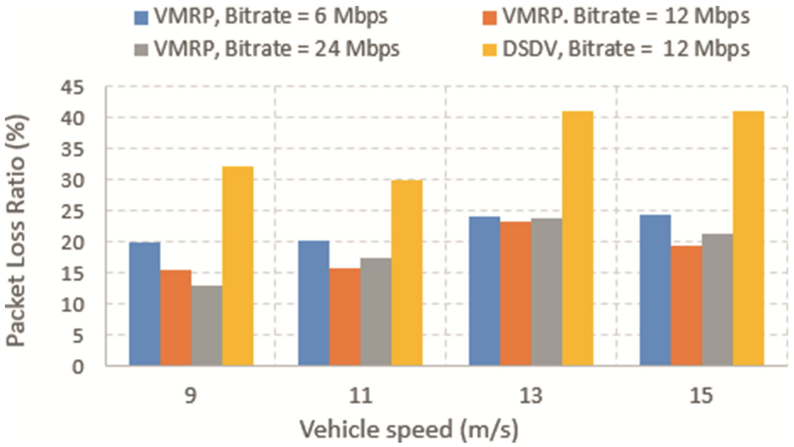


Fig. 10. Effect of vehicle speed to packet loss ratio at different bitrates.

4 Conclusions and Outlook

With the proposed VMRP, it is believed that VANET using this routing strategy will have the capability of supporting low latency, high data rate, and acceptable packet loss rate while keeping the dynamics in traffic movement with high vehicle speed. Besides, this protocol also supports both V2V and V2I communications with stable radio links based on the prediction of vehicles' directions. The simulation results shows that the proposed VMRP outperforms DSDV protocol in terms of packet loss, and data rate.

Prediction of vehicles' stable links when all vehicles are moving with different velocities is also a challenge for the next future works. This problem can be overcome by using MORP to estimate the communication time between vehicles belonging to one group which is successfully performed by VMRP.

Acknowledgement. This research is funded by Ho Chi Minh City University of Technology within the Science Research Project, under grant number #T-DDT-2016-25.

References

1. Karagiannis, G., Altintas, O., Ekici, E., Heijenk, G., Jarupan, B., Lin, K., Weil, T.: Vehicular networking: a survey and tutorial on requirements, architectures, challenges, standards and solutions. IEEE Commun. Surv. Tutorials **13**(4), 584–616 (2011)
2. Hess, S., Segarra, G., Evensen, K., Festag, A., Weber, T., Cadzow, S.: Intelligent transport systems. In: Results from ETSI TC ITS and WG Meetings, 8–12 April 2013
3. Figueiredo, L., Jesus, I., Tenreiro Machado, J.A., Ferreira, J.R., Martins de Carvalho, J.L.: Towards the development of intelligent transportation systems. In: Proceedings of IEEE Intelligent Transportation Systems, Oakland (CA) USA (2001)
4. ITS Japan: ITS Green Safety Showcase. ITS World Congress Tokyo, 14–17 October 2013. http://www.its-jp.org/english/files/2013/10/Intorduction_ITS-GREEN-SAFETY-Dec2013.pdf
5. ITS Japan: Smartway with ACC/CACC (I2V, V2V). 20th ITS World Congress Tokyo (2013). http://www.its-jp.org/english/files/2013/10/Smartway-with-ACCCACC-leaflet.pdf
6. Fukushima, M., Kamata, K., Tsukada, N.: Progress of V-I Cooperative Safety Support System, DSSS, in Japan. IT&ITS Engineering Department, NISSAN MOTOR CO., Ltd. UTMS Society of Japan (2013). http://www.utms.or.jp/
7. Guo, J.: Vehicle safety communications in DSRC. In: US Army 6th Winter Workshop (2006)
8. Barba, C. T.: Contribution to design a communication framework for vehicular ad hoc networks in urban scenarios. Doctor thesis of Philosophy in Telematics in the Department of Telematics Engineering, Barcelona, pp. 5–37 (2013)
9. Hartenstein, H., Laberteaux, K.P.: VANET: Vehicular Applications and Inter-Networking Technologies, 1st edn. Wiley, Hoboken (2010)
10. Wilson, R.: Propagation Losses Through Common Building Materials, pp. 5–16. Magis Networks Inc., San Diego (2002)
11. Li, B., Mirhashemi, M.S., Laurent, X., Gao, J.: Wireless access for vehicular environments. Project report, Department of Computer Science and Engineering, Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden (2013)
12. Li, Y.: An overview of the DSRC/WAVE technology. In: Singh, K., Awasthi, A.K. (eds.) QShine 2013. LNICSSITE, vol. 115, pp. 544–558. Springer, Heidelberg (2012). doi: 10.1007/978-3-642-29222-4_38
13. IEEE 802.11 Working Group: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11™-2 2012. IEEE Standards Association (2012). http://standards.ieee.org/about/get/802/802.11.html. Accessed 29 March 2012
14. Perkins, C., Belding-Royer, E., Das, S.: Ad hoc on-demand distance vector (AODV) routing. IETF RFC **3561**, 2–27 (2003)

15. Johnson, D.B., Maltz, D.A., Broch, J.: DSR: the dynamic source routing protocol for multi-hop wireless ad hoc networks. Monarch Project at Carnegie Mellon University, Pittsburgh, PA, 15213-3891, pp. 3–21 (2001)
16. IETF, RFC 3626. Optimized Link State Routing Protocol (OLSR)
17. Santivanez, C., Ramanathan, R.: Hazy sighted link state (HSLS) routing: a scalable link-state algorithm. BBN technical memo BBN-TM-1301, BBN Technologies, Cambridge, MA, August 2001
18. Bellur, B., Ogier, R.G., Templin, F.L.: Topology Broadcast based on Reverse Path Forwarding (TBRPF). Internet Draft. https://tools.ietf.org/html/draft-ogier-manet-tbrpf-00
19. He, G.: Destination-Sequenced Distance Vector (DSDV) Protocol. Networking Laboratory Helsinki University of Technology (2002)
20. Perkins, C.: Highly Dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In: Proceedings of ACM SIGCOMM 1994, London, UK (1994)
21. Beijar, N.: Zone routing protocol (ZRP). Networking Laboratory, Helsinki University of Technology, P.O. Box 3000, FIN-02015 HUT, Finland (2002)
22. Morris, R., Jannoti, J., Kaashoek, F., Li, J., Decouto, D.: CarNet: a scalable ad-hoc wireless network system. In: Proceedings of the 9th ACM SIGOPS European Workshop, Kolding, Denmark (2000)
23. Menouar, H., Lenardi, M., Filali, F.: A movement prediction based routing protocol for vehicle-to-vehicle communications. In: Proceedings of V2VCOM 2005, San Diego, USA (2005)
24. Morris, R., Kaashoek, F., Karger, D., Aguayo, D., Bicket, J., Biswas, S., Couto, D.D., Li, J.: Grid: scalable ad-hoc wireless networking. The Grid Ad Hoc Networking Project, Massachusetts Institute of Technology University
25. Online: OMNeT++ Network Simulation Framework. OMNeT++ Community (2013). http://www.omnetpp.org/
26. Online: INET Framework for OMNeT++. OMNeT++ Community (2012). http://inet.omnetpp.org/
27. VEINS. The open source vehicular network simulation framework. http://veins.car2x.org/
28. Online: SUMO – Simulation of Urban Mobility. German Aerospace Center, Institute of Transportation Systems (2014). http://sumo-sim.org/
29. Booysen, T.: Tutorial: Simulating VANET and ITS (using OMNeT++ and SUMO). Seminar at UniRC (2012)

Heuristic-Guided Verification for Fast Congestion Detection on Wireless Sensor Networks

Khanh Le[1(✉)], Toan Nguyen[1], Thanh Cao[2], Thang Bui[1], and Tho Quan[1]

[1] Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
{lnkkhanh,thang,qttho}@cse.hcmut.edu.vn,
51103696@hcmut.edu.vn
[2] Saigon University, Ho Chi Minh City, Vietnam
ctpthanh@sgu.edu.vn

Abstract. Petri Net (PN) are widely used to model distributed systems due to its powerful capability of simulating stepwise behaviors of the systems in both sequential or concurrent manners. Furthermore, PN models can be formally verified of their properties using model checking. However, when applied in practical situations, this approach suffers from the infamous problem of state space explosion. In this paper, we suggest a heuristic approach which can potentially reduce the resource consumed by the verification process on a PN-modeled system. We illustrate our idea by an application of congestion detection of Wireless Sensor Networks (WSN), once represented as PN models. The experimental results confirm the improvements gained by our approach.

Keywords: Wireless sensor networks · Petri nets · Heuristic-guided verification

1 Introduction

The Petri Net (PN) modeling languages are widely used in research and industry communities. There are several tools developed to help users specify and verify PN models, in particular Snoopy [1], TAPAAL [2], CosyVerif [3] or CPN Tools [4].

The PN formalism [5] is a graphical mathematical language which efficiently supports the modeling and verification of distributed systems. Basically, a PN is a directed bipartite graph, featuring transitions and places. Each *place* contains a number of *tokens*. Any distribution of tokens over the places will represent a configuration of the net called a *marking*. When all source *places* of a *transition* have enough requested *tokens*, this *transition* is *firable*. Whenever the *transition* is *fired*, the requested *tokens* are removed from the source *places* and new *tokens* are created in destination *places*. In the other hand, when the tokens are moved, the new markings will be created, and the set of all markings will form the *state space* or a directed graph whose nodes represent reachable states of the system

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 105–116, 2016.
DOI: 10.1007/978-3-319-48057-2_7

and the arcs represent the transitions between states. This graph is also called *reachability graph* (RG).

The process of checking one property on models is actually done by using Model Checking (MC) technique. This technique will verify whether a property holds on a model by exploring all of possible states in RG. Based on how exploring the RG, MC techniques fall into two categories including *explicit MC* and *implicit MC* [6]. Explicit MC technique explores explicitly all states whereas the other one explores a set of states in a single step instead of enumerating one state at a time.

In turn, the technique that traverses explicitly these nodes on RG is also divided into two kinds based on the way of state space's building including [7]

1. technique that builds the state space *completely* then start exploring all reachable states
2. *on-the-fly* technique that builds the state space dynamically on-demand during the MC operations.

In order to check whether properties are satisfied, MC uses a simple (Depth-First or Breadth-First) Search to explore all states on RG from the initial state and try to reach some targets. If the target cannot be reached on one path even though the end state is visited, the algorithm will return and choose another one. It is easy to recognize that there are a huge number of paths which do not lead to target in RG, thus leading to the famous state space explosion situation.

In order to prevent the state space explosion, we propose a oriented verification algorithm based on the spirit of heuristic, called HGV-WSN (Heuristic-Guided Verification for WSN). This algorithm leads the search algorithm of MC to the target state quickly. The oriented verification of HGV-WSN is done by using the knowledge which comes from *heuristic table*. This table is constructed based on the *minimal state space* and promotes its power when explore on-the-fly *real state space* or *real RG*. Owing to be controlled by heuristic, the search will skip the unreachable-target paths and thus coming the target quickly. Note that, the number of states in *minimal state space* is finite whereas is infinite in the real one. Such contribution makes our approach become an extraordinary.

In order to easy illustrate the working of our algorithm, we use a model whose name is WSN-PN [8]. This tool represents a wireless sensor network (WSN) model by using PN language. Congestion detection became the main property is checked on such model. The result of this tool can show whether a congestion occurs on WSN. However, the number of states of RG is too huge. Thus, applying our proposal is an promised solution for such problem.

Outline. The rest of the paper is organised as follows. Section 2 reviews congestion detection techniques on WSN Model. After that, Sect. 3 explains detailed our algorithm. More extensive experiments are reported in Sect. 4. Then, Sect. 5 draws conclusions and outlines future work. Finally, Sect. 6 discusses related works.

2 Congestion Detection on PN-Modeled Using WSN-PN

This section is a short introduction about the activity of congestion detection on WSN model by using WSN-PN model. Firstly, WSN-PN is a WSN PN-Modelled, after that the verification is processed on model to detect the congestion.

2.1 Petri Net Generation for a WSN

A WSN consists of several *sensors* that can communicate with each other using WiFi signals, *i.e.* $WSN = \{S, C\}$, where S is the set of sensors and C is the set of channels. There are three types of sensors: *source*, *sink* and *intermediate node.* The role of intermediate nodes is to receive and forward packets, as depicted in the oil monitoring application reported by [9]. In some applications, the role of source and intermediate sensors are the same, *i.e.*they both can generate and send packets. In that case, it could be modeled as a combination of a source node and an intermediate node in WSN-PN.

Sensors can be connected in *unicast*, *multicast* or *broadcast* mode, each of which specifies whether certain pairs of sensors can exchange information or not. If two sensors can communicate, there is a *channel* established between them. Information on sensors and channels forms the topology of a WSN. An example of WSN topology is given in Fig. 1. Sensors play the role of intermediate nodes, conveying information from a source (denoted by a double-lined circle) to a sink (denoted by a full circle).

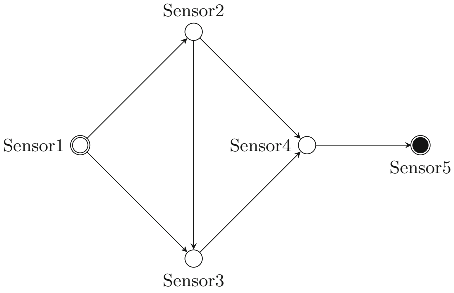
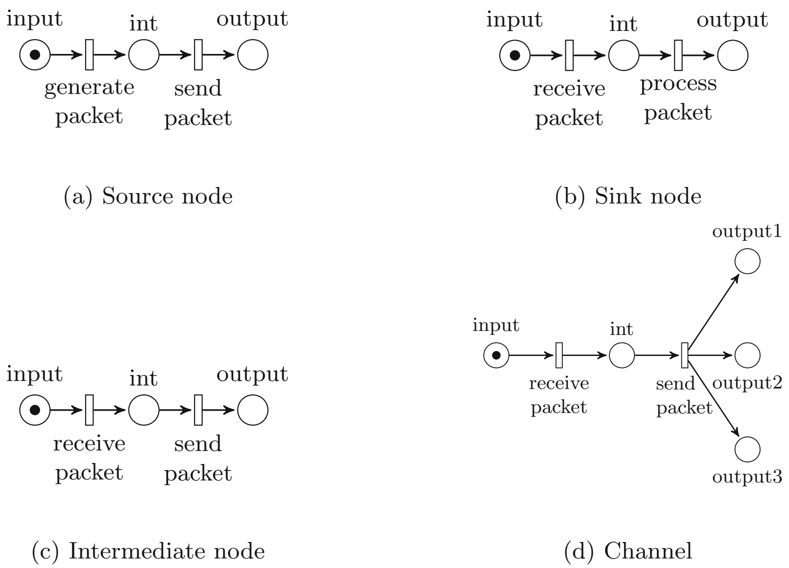


Fig. 1. Wireless sensor network topology

To conduct a Component-based Petri net modelling approach from a WSN, sensors and channels are first modelled individually as *Component Petri nets.* Then, these Component Petri Nets are combined together, forming the final Component-based Petri Net of the WSN as depicted in Fig. 2.

It is not only necessary to represent the flow in the WSN as a PN, but also to attach to transitions some code that manipulates other information such as *sensor buffer size* or information concern to the terms of *timing* such as *sending rate* and *processing rate.* The processing rate indicates the number of packets a



(a) Source node

(b) Sink node

(c) Intermediate node

(d) Channel

Fig. 2. Component Petri net models of sensors and channels

sensor can handle (transfer from buffer to queue) over a given period of time, while the sending rate specifies the number of packets sent by a sensor to its connected channels (the detailed of such parameters is noted at [8]).

2.2 Congestion Detection

We use the PAT model-checker [10] to verify the following LTL property on the PN model of a WSN that expresses congestion:

```
#assert WSN() |= []<> Congestion
```

where `[]<> Congestion` stands for the LTL operations of $\Box\Diamond$ (which means *always eventually*) and the condition whether a `Congestion` occurs or not. In our WSN-PN tool, the valuation of whether `Congestion` holds or not at a certain state is simulated by *C#* code as follows. To detect congestion on a sensor, our simulated code counts the number of received packets at the sensor. If this number reaches a threshold (*i.e.* greater than 70 % buffer size, based on [11] conclusions), the guard condition lets the flow reach a particular state making `Congestion` hold. A similar method is applied for detecting congestion in channels.

The *C#* source code to check whether a sensor's buffer is full (*i.e.* causing congestion) is as follows:

```
public bool isFullSensor(int id){
    return(sensors[id].PBuffer.Count >=
        sensors[id].BufferMaxSize);
}
```

3 State Space Reduction by Using Heuristic Search

By adopting the idea of the famous A* search algorithm, we propose an heuristically verification algorithm whose purpose is finding a next suitable node on RG. For each step, the decision of next node is done based on an evaluated function which we called f. The value of f is calculated as $f = g + h$ where g is the visited distance from the initial node to present node and h is the distance from the present node to target. The value of g and h are also based on a heuristic table which is constructed on *minimal state space*. Due to the orientation of this evaluated function f, the number of visited states when exploration is reduced significantly.

We used WSN-PN to implement this proposal. The implementation is divided to 3 steps including

- generating *minimal state space*;
- constructing heuristic table;
- applying evaluated function f to find a suitable path on *real state space*.

Each step will be clearly describe in next subsections.

3.1 Minimal State Space Construction

Minimal state space or *minimal reachability graph* is constructed when we let every enable transition fire once. To do so, the parameters including the number of packets, the sending rate and the processing rate is set equal to 1. RG is built based on [12].Let start an example in Fig. 1. The corresponding PN is generated from this network topology in Fig. 3.

Starting from place $Main1$, the token will be move to place $Output1$ due to the fired transition $Send1$. The graph is created with root is node 1 and the arc which the name is $Send1$. Continuing with fired transitions $Channel1 - 2$ and $Channel1 - 3$, we have one path from 1 to 2 whose name is $Channel1 - 2$ and another one from 1 to 3 whose name is $Channel1 - 3$. Following this way, a reachbility graph is created. Note that, WSN-PN using the on-the-fly method to generate RG. So, whenever reach Congestion node, the algorithm will stop. Assume Congestion node here is $Congestion4$, corresponding to node 9 in a part of minimal RG shown in Fig. 4.

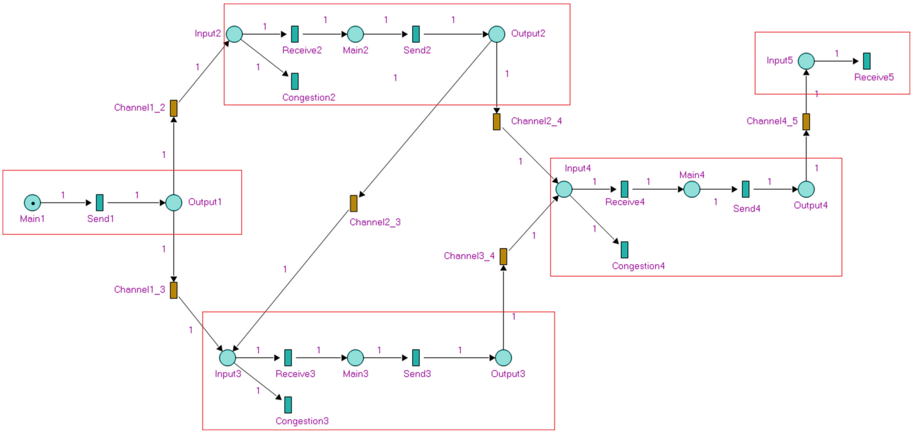


Fig. 3. PN model of Fig. 1

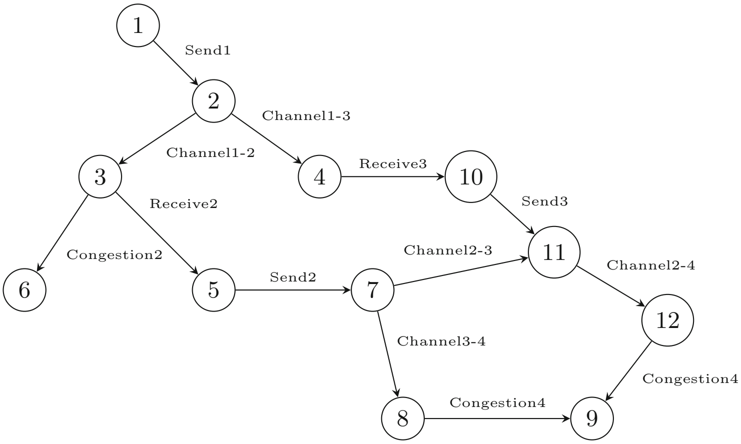


Fig. 4. Inital state space example

3.2 Heuristic Table Construction

As discussed, evaluated function $f(x)$ will use information which is supplies by heuristic table to explore one path on real state space in next step. Heuristic table is constructed based on the minimal RG in previous step. Each line in this table contains all paths from the initial node to one target. Targets in this paper are assumed as *Congestion* nodes. In turn, one path contains the number of nodes from the target to any nodes in the graph. In order to find all paths, we use a reversed-BFS algorithm where source is the target and destination is the initial node.

From the minimal RG on Fig. 4, we have an example of heuristic table with the target *Congestion*4 in Table 1.

Table 1. Heuristic table generating

Target						
Congestion 4 (Node 9)	**Node** **Path**	2	3	..	11	12
	Path 1	6	5	..	0	0
	Path 2	6	0	..	2	1
	Path 3	7	6	..	2	1

3.3 Real State Space Construction Based on Heuristic Table

Exploration on real state space is more complicated than initial state space even though using on-the-fly technique. In minimal state space, each transition is just fired once. However, in reality, due to the value of sending rate on sensors and the number of packets, each transition may be fired much more than once. For instance, *Sensor*1 wants to send 10 packets to *Sensor*3, it will send 5 packets at the first time and the rest for the second one due to its sending rate is 5 packets/s. Thus, based on the PN model in Fig. 3, transition $Channel1 - 3$ is fired twice on the same marking, corresponding to twice sending times.

Figure 5 is a example of RG which corresponds to minimal RG of Fig. 4.

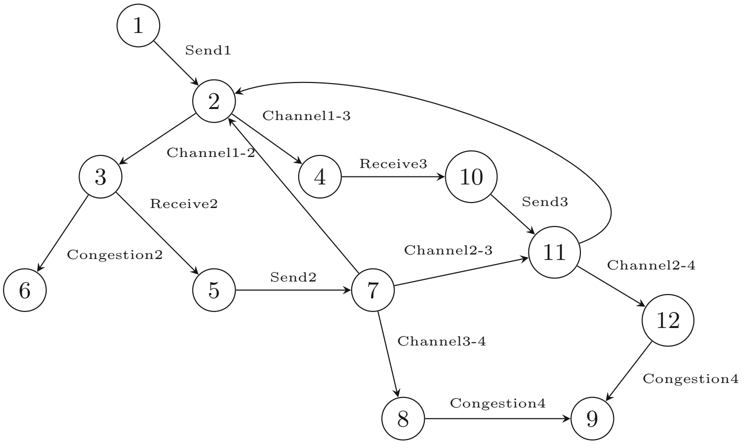


Fig. 5. Example of real state space of Fig. 4

It is easy to recognize that we have a cycle line from *state*11 to *state*2 due to the twice firing. If this circle is passed several times, the traversed path will be longer and thus wasting verification or even though leading to the state space explosion problem.

This example just shows an small part of real RG. The more number of sensors or packets as well as the smaller sending rate, the more complicated real reachibility graph. For detailed, real state space is captured by the tool in Fig. 6.

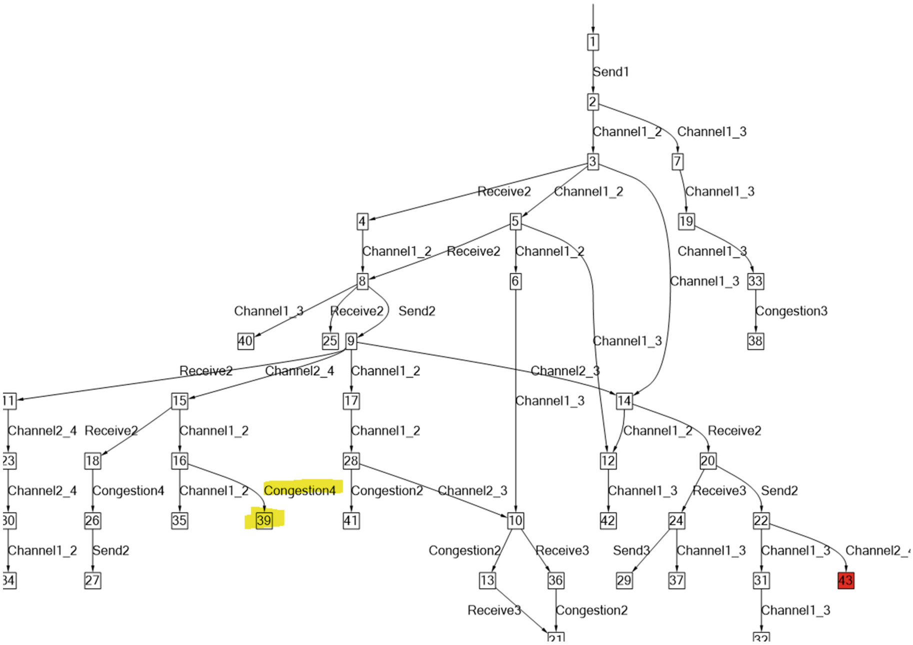


Fig. 6. Real state space of Fig. 4

Based on the knowledge which extract from heuristic table and the working of on-the-fly technique, this proposal will construct a real state space which allowing the search algorithm of MC can reach the target faster. To do so, we use the evaluated function f which is a guidance for each node to find it next node in the graph. f is calculated by $f = g + h$ where g is the number of visited node from the initial node to present node and h is the number of nodes will be visited from the present node to target. The value of h is extract from constructed heuristic table in the previous task. g is calculated during the traversing process from the initial node to the present one.

Let see example in Table 2 for seeing the execution of evaluated function.

Table 2. Working of evaluated function

Node	Next node	$f = g + h$	Path
1	2	$f = 1 + 6 = 7$	$1 \rightarrow 2$
2	3 4	$f = 2 + 5 = 7$ $f = 2 + 5 = 7$	$1 \rightarrow 2 \rightarrow 3$
..			
11	12	$f = 5 + 2 = 7$	$1 \rightarrow 2 \rightarrow 3 \rightarrow ... \rightarrow 11 \rightarrow 12$
	1	$f = 5^n + 2$	

4 Experimentation

We conducted experiments to demonstrate the efficiency of our heuristic algorithm, which can significantly reduce the visited state space of congestion verification. The experiments were ran using WSNs modelled by WSN-PN, whose numbers of sensors range from 1 to 15. The parameters of these sensors are set to enforce the congestion. Table 3 illustrates the main parameters of a WSN.

Table 3. Initial parameters

Parameter	Range
Sensor buffer size	200–600
Channel bandwidth (buffer)	200–600
Number of packets	30–50 packets/s
Processing rate	1–2 packets/s
Transfer rate	1–2 packets/s

We also compare the result of congestion detect in case of using heuristic search and a normal case with BFS in Table 4.

Based on the result, the results of congestion detection of both algorithms are the same, *i.e.* they are both *valid* (congestion) or *not valid* (non congestion). Moreover, most of cases, the number of visited states of our propose is reduced significantly, except the case of *deadlockfree* property. With PN model, *deadlockfree* means that each reachable marking enables a transition [13]. The algorithm must examine all paths in RG and thus, the number of visited state is equal to normal one.

Table 4. Experimental results

Number of Sensors	Number of Packets	Bandwidth / Buffer	Sending Mode	Model	Property	Visited states BFS	Result BFS	Visited states Heuristic	Result Heuristic
5	10	50	Broadcast	No Abstraction	*deadlockfree*	614	Valid	614	Valid
					chk-channel-congestion	90	Not valid	31	Not valid
					chk-sensor-congestion	32	Not valid	9	Not valid
				Channels Abstraction	*deadlockfree*	297	Valid	297	Valid
					chk-sensor-congestion	20	Not valid	7	Not valid
				Sensors Abstraction	*deadlockfree*	114	Valid	114	Valid
					chk-channel-congestion	56	Not valid	21	Not valid
5	10	50	Multicast	No Abstraction	*deadlockfree*	14	Valid	14	Valid
					chk-channel-congestion	24	Not valid	10	Not valid
					chk-sensor-congestion	16	Not valid	6	Not valid
				Channels Abstraction	*deadlockfree*	9	Valid	9	Valid
					chk-sensor-congestion	12	Not valid	5	Not valid
				Sensors Abstraction	*deadlockfree*	1125	Valid	1125	Valid
					chk-channel-congestion	22	Not valid	22	Not valid
5	10	50	Unicast	No Abstraction	*deadlockfree*	14	Valid	14	Valid
					chk-channel-congestion	24	Not valid	10	Not valid
					chk-sensor-congestion	16	Not valid	6	Not valid
				Channels Abstraction	*deadlockfree*	5	Valid	5	Valid
					chk-sensor-congestion	9	Not valid	5	Not valid
				Sensors Abstraction	*deadlockfree*	8	Valid	8	Valid
					chk-channel-congestion	14	Not valid	7	Not valid
10	10	50	Broadcast	No Abstraction	*deadlockfree*	28700	Valid	28700	Valid
					chk-channel-congestion	138	Not valid	46	Not valid
					chk-sensor-congestion	219	Not valid	38	Not valid
				Channels Abstraction	*deadlockfree*	11346	Valid	11346	Valid
					chk-sensor-congestion	118	Not valid	28	Not valid
				Sensors Abstraction	*deadlockfree*	2826	Valid	2826	Valid
					chk-channel-congestion	87	Not valid	32	Not valid
10	10	50	Multicast	No Abstraction	*deadlockfree*	39	Valid	39	Valid
					chk-channel-congestion	39	Not valid	14	Not valid
					chk-sensor-congestion	54	Not valid	20	Not valid
				Channels Abstraction	*deadlockfree*	27	Valid	27	Valid
					chk-sensor-congestion	36	Not valid	15	Not valid
				Sensors Abstraction	*deadlockfree*	36750	Valid	36750	Valid
					chk-channel-congestion	26	Not valid	9	Not valid
10	10	50	Unicast	No Abstraction	*deadlockfree*	39	Valid	39	Valid
					chk-channel-congestion	39	Not valid	14	Not valid
					chk-sensor-congestion	54	Not valid	20	Not valid
				Channels Abstraction	*deadlockfree*	3	Valid	3	Valid
					chk-sensor-congestion	6	Valid	6	Valid
				Sensors Abstraction	*deadlockfree*	21	Valid	21	Valid
					chk-channel-congestion	26	Not valid	9	Not valid
15	10	50	Broadcast	No Abstraction	*deadlockfree*	Time out at 157.863		Time out at 153.233	
					chk-channel-congestion	840	Not valid	42	Not valid
					chk-sensor-congestion	1217	Not valid	1217	Not valid
				Channels Abstraction	*deadlockfree*	Time out at 156.6		Time out at 153.823	
					chk-sensor-congestion	556	Not valid	556	Not valid
				Sensors Abstraction	*deadlockfree*	Time out at 133.62		Time out at 135.262	
					chk-channel-congestion	489	Not valid	29	Not valid
15	10	50	Multicast	No Abstraction	*deadlockfree*	65	Valid	65	Valid
					chk-channel-congestion	69	Not valid	13	Not valid
					chk-sensor-congestion	70	Not valid	16	Not valid
				Channels Abstraction	*deadlockfree*	42	Valid	42	Valid
					chk-sensor-congestion	44	Not valid	13	Not valid
				Sensors Abstraction	*deadlockfree*	13739	Valid	13739	Valid
					chk-channel-congestion	47	Not valid	8	Not valid
15	10	50	Unicast	No Abstraction	*deadlockfree*	65	Valid	65	Valid
					chk-channel-congestion	69	Not valid	13	Not valid
					chk-sensor-congestion	70	Not valid	16	Not valid
				Channels Abstraction	*deadlockfree*	4	Valid	4	Valid
					chk-sensor-congestion	8	Valid	8	Valid
				Sensors Abstraction	*deadlockfree*	37	Valid	37	Valid
					chk-channel-congestion	43	Valid	8	Valid

5 Conclusion

This paper presents a heuristic search algorithm for verifying congestion on Wireless Sensor Networks using Petri nets. This algorithm is a guidance for the state space generation of MC based the evaluated function. Evaluated function is calculated by each step based on the heuristic table which is constructed in the minimal state space. The experimental results prove the efficiency of algorithm thanks to the number of visited states.

The heuristic algorithm is now just based on the distance of sensor, in the future, we will concentrate more parameters that can be affected the congestion such as the energy of sensors or environment factors.

6 Related Works

6.1 Network Modelling Using Petri Nets

Petri nets are particularly well-suited for modelling and analysing network systems and protocols [14]. In [15] a model is proposed to implement ATM (Asynchronous Transfer Mode) networks and the applications running over it. Transferring all multimedia data over the switching network of an ATM network constitutes a big challenge. Since ATM is a connection oriented protocol, the switching network must establish a virtual connection from one of its input ports to an output port before forwarding incoming ATM cells (packets) along that virtual connection. Because of restricted bandwidth, multimedia data must be split to fix-length units before sending without losing behaviour. Most approaches use queue-based or stochastic-based models, but however ignore synchronisation. Dividing the ATM network into some synchronous models and forcing them to cooperate increases the transmission rate. Using Petri nets for modelling has proved promising.

[16] uses Petri nets to model a LAN switched network architecture. Components of this model include switches, servers, clients and interaction between them. The purpose of this model is to verify the influence of the switch buffer size and the rate of packets loss on the quality of the transmission.

Stochastic Petri nets are also used to model and analyse ad-hoc wireless networks in [17]. Ad-hoc wireless networks are dynamic networks with mobile nodes whose bandwidth and battery are limited.

Acknowledgment. This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.01-2015.16.

References

1. Heiner, M., Richter, R., Schwarick, M.: Snoopy: a tool to design and animate/simulate graph-based formalisms. In: Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems and Workshops (SimuTools 2008), p. 15 (2008)

2. Byg, J., Jørgensen, K.Y., Srba, J.: An efficient translation of timed-arc petri nets to networks of timed automata. In: Breitman, K., Cavalcanti, A. (eds.) ICFEM 2009. LNCS, vol. 5885, pp. 698–716. Springer, Heidelberg (2009)
3. André, É., Lembachar, Y., Petrucci, L., Hulin-Hubard, F., Linard, A., Hillah, L., Kordon, F.: Cosyverif: an open source extensible verification environment. In: 18th International Conference on Engineering of Complex Computer Systems (ICECCS 2013), pp. 33–36 (2013)
4. Westergaard, M., Slaats, T.: CPN tools 4: a process modeling tool combining declarative and imperative paradigms. In: 11th International Conference on Business Process Management (BPM 2013), pp. 393–402 (2013)
5. Kozura, V.E., Nepomniaschy, V.A., Novikov, R.M.: Verification of distributed systems modelled by high-level Petri nets. In: 2002 International Conference on Parallel Computing in Electrical Engineering (PARELEC 2002), pp. 61–66 (2002)
6. Garavel, H., Mateescu, R., Smarandache, I.M.: Parallel state space construction for model-checking. In: Dwyer, M.B. (ed.) SPIN 2001. LNCS, vol. 2057, p. 217. Springer, Heidelberg (2001)
7. Barnat, J., Brim, L., Rockai, P.: On-the-fly parallel model checking algorithm that is optimal for verification of weak LTL properties. Sci. Comput. Program. **77**(12), 1272–1288 (2012)
8. Le, K., Bui, T., Quan, T., Petrucci, L., André, É.: Congestion verification on abstracted wireless sensor networks with wsn-pn tool. Adv. Comput. Netw. **4**(1), 33–40 (2016)
9. Luo, Y., Lina, P., Zuba, M., Peng, Z., Cui, J.-H.: Challenges and opportunities of underwater cognitive acoustic networks. IEEE Trans. Emerg. Top. Comput. **2**(2), 198–211 (2014)
10. Si, Y., Sun, J., Liu, Y., Dong, J.S., Pang, J., Zhang, S.J., Yang, X.: Model checking with fairness assumptions using PAT. Front. Comput. Sci. **8**(1), 1–16 (2014)
11. Wan, C.-Y., Eisenman, S.B., Campbell, A.T.: CODA: congestion detection and avoidance in sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys 2003), pp. 266–279. ACM (2003)
12. Mayr, E.W.: An algorithm for the general petri net reachability problem. SIAM J. Comput. **13**(3), 441–460 (1984)
13. Proth, J.-M., Xie, X.: Petri Nets: A Tool for Design and Management of Manufacturing Systems. Wiley, New York (1996)
14. Billington, J., Wheeler, G.R., Wilbur-Ham, M.C.: PROTEAN: a high-level Petri net tool for the specification and verification of communication protocols. IEEE Trans. Softw. Eng. **14**(3), 301–316 (1988)
15. Reid, M., Zuberek, W.M.: Timed Petri net models of ATM LANs. In: Billington, J., Diaz, M., Rozenberg, G. (eds.) Application of Petri Nets to Communication Networks. LNCS, vol. 1605, pp. 150–175. Springer, Heidelberg (1999). doi:10.1007/BFb0097776
16. Zaitsev, D.A.: Switched LAN simulation by colored Petri nets. Math. Comput. Simul. **65**(3), 245–249 (2004)
17. Zhang, C., Zhou, M.: A stochastic Petri net-approach to modeling, analysis of ad hoc network. In: Information Technology: Research and Education (ITRE 2003), pp. 152–156. IEEE (2003)

Security and Privacy Engineering

Security Risk Management in the Aviation Turnaround Sector

Raimundas Matulevičius[1], Alex Norta[2](✉), Chibozur Udokwu[2], and Rein Nõukas[2]

[1] Institute of Computer Science, University of Tartu, J. Liivi 2, 50409 Tartu, Estonia
rma@ut.ee

[2] Department of Informatics, Tallinn University of Technology, Akadeemia Tee 15A, 12816 Tallinn, Estonia
alex.norta.phd@ieee.org, cjobuzor@gmail.com, rein.noukas@gmail.com

Abstract. Security in the airline industry receives heightened attention due to an increase of diverse attacks, many being driven by information technology. Ongoing research does not take into account the sociotechnical nature of security in critical domains such as airline turnaround systems. To cut time and costs, the latter comprises several companies for ticket- and luggage management, maintenance checks, cleaning, passenger transportation, re-fueling, and so on. The airline industry has adopted extensively information technology for assuring an incoming airplane is in a state to take off again as quickly as possible. Increasingly, this leads to the emergence of a virtual enterprise that uses information technologies to seamlessly integrate respective airline-turnaround processes into one composition. The resulting sociotechnical security risk management issues are not well understood and require diligent investigation. This paper fills the gap with an evaluation about the application of a security risk management method to identify critical business- and information-technology assets for a deeper risk mitigation analysis. The results of this paper yield insights about the utility of existing security risk management approach.

Keywords: Security · Risk analysis · Airline turnaround · Virtual organization · Decentralization · Composition · Mitigation · Sociotechnical · E-governance · Business process · Cross-organizational

1 Introduction

The airline industry experiences a profound penetration with information technology in all aspects [5]. Consequently, many novel risks and security issues are associated with civil aviation that result in worst cases with catastrophic crashes of airlines. Other critical security issues are related to communication, such as a deliberate jamming of Automatic Dependent Surveillance-Broadcast (ADS-B) systems [8] that are a surveillance technology in which an aircraft determines its position via satellite navigation and periodically broadcasts it for tracking

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 119–140, 2016.
DOI: 10.1007/978-3-319-48057-2_8

by air traffic control ground stations. Furthermore, the recognition emerges that the aviation industry turns rapidly into a cyber-physical system (CPS) [19] that poses additional risks and security issues. Briefly, a CPS [4] is a system composed of physical entities that are controlled or monitored by computer-based algorithms.

While the initial approach to studying airport-related security is rather technical, recent work acknowledges that this is a socio-technical systems [9] matter. The latter is characterized by a complex organizational work design where people solve problems at their workplaces with the means of rather sophisticated technology. In [10], the authors recognize for the first time the socio-technical nature of airports by using extended use-case diagrams and storyboard representations of use cases to discover stakeholder requirements such as security for the development of an airport operating system. Security in connection with information is explicitly investigation in [6] that compares practitioner-oriented risk management methods and several academic security modeling frameworks to develop an ontology, or domain model, of information system security risk management. More recently, in [11] the authors investigate requirements evolution in the context of the SecureChange[1] EU-project in which the industry case is drawn from the Air Traffic Management (ATM) domain. While safety- and security experts are part of the focus groups, the case study results do not explicitly zoom into security specifics in their study results. Furthermore, parameter measurability and social aspects of security policies in [20] investigate specifically the costs versus benefit trade-off in alternative airport security policies constellations pertaining to, e.g., passengers, items such as baggage, and so on.

Literature shows security-focused research for airline management is a topical area of interest. However, the security topics are very specific and do not acknowledge that modern information technology enables ad-hoc and process-aware cross-organizational collaborations [7,15–17] that benefit significantly the reduction of time and costs of airline management while yielding simultaneously improvements in quality. Such novel ways of airline management systems also lead to unusual security risk issues for which the mitigation strategies are unclear. This paper fills the gap with an evaluation that answers the research question of *how to use information system security risk management* (ISSRM) *in cross-organizational collaborations* (COC) *of the airline industry to achieve a desired level of security.* For establishing a separation of concerns, we deduce the following sub-questions: What are relevant assets in airline COC that need to be secured? What security risks threaten COC systems? What are security risk mitigation strategies in airline COC?

The remainder of this paper structured as follows. Section 2 gives background information that is necessary for being able to follow the rest of the paper. Section 3 gives essential properties that characterize agent negotiation. Next, Sect. 4 finds security risks that threaten airline turnaround systems. Section 5 gives mitigation strategies to protect from security risks and Sect. 6 shows a security trade-off analysis process. Finally, Sect. 7 concludes with important lessons learned and future work directions.

[1] http://www.securechange.eu/.

2 Background

We give additional information that is relevant for the remainder of this paper. First, Sect. 2.1 comprises a running case that stems from studying COC in an airline-turnaround scenario. Secondly, Sect. 2.2 shows facets of a security risk management framework that is relevant for studying our running case.

2.1 Running Airline-Turnaround Case

We use the business process model notation BPMN 2.0 [13] for showing the running case in Fig. 1. BPMN is an industry standard for business-process modeling that provides a graphical notation showing process steps as boxes of various kinds and their order by connecting them with arrows.

The airline-turnaround process in Fig. 1 is taken from a larger model in [14] and shows three swimlanes for ground services, passenger management and gate agent respectively. The swimlane for ground services commences with a start signal event to begin after-flight services, followed by an AND-split. The top parallel branch has a catching intermediary start signal event for when all passengers have de-boarded, followed by yet another AND-split parallel gateway for cleaning, restocking aircraft, and fueling after a start message event indicates the receipt of a fuel slip. The task for restocking the aircraft requires a data object comprising passenger information, e.g., about specific dietary needs. After an AND-join, an intermediate signal event informs that boarding is now allowed. The bottom branch of the initial AND-split begins with a task for offloading cargo and luggage, followed by an AND-split with parallel branches and respective intermediate message event nodes. The top parallel branch waits via a catching intermediate event for a message from an adjacent process that indicates a cargo assignment and the second parallel branch likewise needs to wait until catching the message that the luggage receipt exists. After the AND-join, cargo- and luggage-loading commences, culminating into another AND-join before an end-signal event terminates the process for ground services.

The middle swimlane of Fig. 1 for passenger management commences with a start timer event, namely, 24 h before the estimated time of departure (ETD). The latter is followed by an AND-split with the top parallel branch starting a sequence with a task for passenger check-ins. The latter takes a data object as input comprising external passenger information and the task also contributes to a data object about checked in passengers. Next in the sequence follows an intermediate timer event to wait until 4 hours before ETD for luggage check-in. The actual task for luggage check-in uses the data object about checked in passengers and produces new facts for a data object abut luggage information. After the completed check-in and given there is one hour left until ETD, an intermediate message event sends information for the ground operations about the luggage being ready to the swimlane for ground services before the AND-join. The bottom parallel branch starts with an intermediate signal event to start after-flight services, followed by a task for conducting the de-boarding of the passenger from the landed airplane. After that, one intermediate signal event

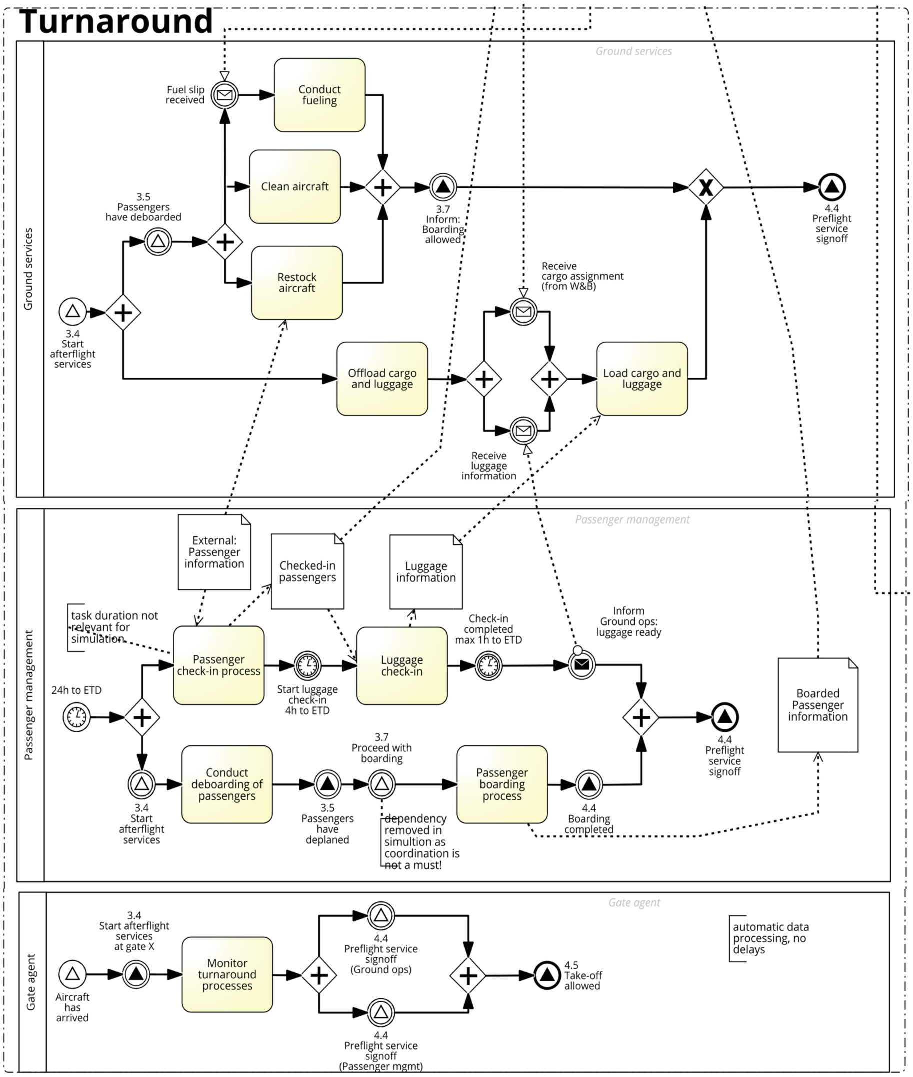

Turnaround
Ground services
Ground services
Fuel slip received
Conduct fueling
Clean aircraft
Restock aircraft
3.5 Passengers have deboarded
3.4 Start afterflight services
3.7 Inform: Boarding allowed
4.4 Preflight service signoff
Receive cargo assignment (from W&B)
Offload cargo and luggage
Load cargo and luggage
Receive luggage information
Passenger management
Passenger management
External: Passenger information
Checked-in passengers
Luggage information
task duration not relevant for simulation
24h to ETD
Passenger check-in process
Start luggage check-in 4h to ETD
Luggage check-in
Check-in completed max 1h to ETD
Inform Ground ops: luggage ready
4.4 Preflight service signoff
Boarded Passenger information
3.4 Start afterflight services
Conduct deboarding of passengers
3.5 Passengers have deplaned
3.7 Proceed with boarding
dependency removed in simultion as coordination is not a must!
Passenger boarding process
4.4 Boarding completed
Gate agent
Gate agent
3.4 Start afterflight services at gate X
Aircraft has arrived
Monitor turnaround processes
4.4 Preflight service signoff (Ground ops)
4.4 Preflight service signoff (Passenger mgmt)
4.5 Take-off allowed
automatic data processing, no delays

Fig. 1. Airline turnaround process, adapted from [14].

indicates all passengers have deplaned, followed by yet another intermediate signal event in the sequence stating that boarding may proceed. The subsequent task for the actual boarding process affects the data object for boarded passenger information. Following the final intermediate signal event about boarding having completed, the AND-join leads to the end signal event for signing off the preflight service.

The final swimlane in Fig. 1 for the gate agent commences with the start signal event that the aircraft has arrived, followed by the intermediate signal event for starting the afterflight services at a specific gate. Furthermore, the gate agent monitors via a task the turnaround process before an AND-split where in parallel two intermediate signal events indicate a preflight service sign-off for ground operation and for passenger management respectively. The subsequent AND-join leads to the end signal event for allowing an airplane takeoff.

2.2 Security Risk Management Domain Model

The ISSRM domain model [6,12] consists of asset-related, risk-related and risk treatment-related concepts. In Fig. 2, this domain model is presented as a UML class diagram for which we briefly present the concepts below.

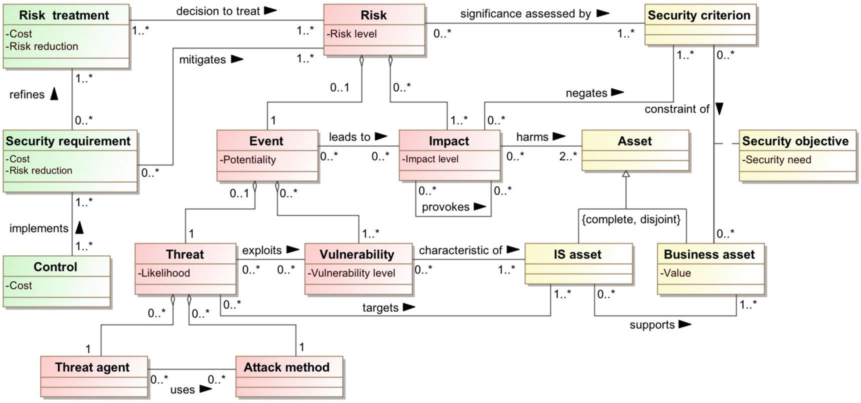


Fig. 2. The ISSRM domain model, adapted from [6,12].

Asset-related concepts describe what organizational assets are important to protect and what criteria guarantee a certain level of asset security. An *asset* is anything of value and that plays a role to accomplish the organization's objectives. Assets can be classified into business assets or organizational assets. A *business asset* describes the information, processes, capabilities and skills essential to the business and its core mission. The *value* metric is used to estimate the security need of each business asset in terms of confidentiality, integrity and availability (see below). An *IS asset* that we later depict as a *system asset* too, is a component or part of an information system that is valuable to an organization as it supports business assets.

A *security criterion* characterizes the security need as a property or constraint on business assets. Thus, the security criterion describes the security needs, which are, typically, expressed through confidentiality, integrity and availability of business assets. A metric to assess a *security need* expresses the importance of security criterion with respect to business asset. This metric is introduced as the attribute to the security objective concept.

Risk-related concepts introduce definitions of risk itself and its immediate components. A *risk* is the combination of a threat with one or more vulnerabilities leading to a negative impact harming at least two or more assets. An *impact* is the potential negative consequence of a risk that negates the security criterion defined for business assets in order to harm these assets when a threat (or an event) is accomplished. A risk *event* is an aggregation of threat and one or more vulnerabilities. A *vulnerability* is the characteristic of an IS asset or group of IS assets that expose a weakness or flaw in terms of security. A *threat* is an incident initiated by a threat agent using an attack method to target one or more IS assets by exploiting their vulnerabilities. A *threat agent* is an agent who has means to intentionally harm IS assets. A threat agent triggers a threat and, thus, is the source of a risk. The threat agent is characterized by expertise, his available resources, and motivation. An *attack method* describes a standard means by which a threat agent executes a threat.

Risk is estimated using a *Risk level* metric. The risk level depends on the event *Potentiality* and the *Impact level*. An event's *Potentiality* depends on the threat *Likelihood* and *Vulnerability level*. It is necessary to note that a threat agent and attack method do not have their own metrics representing their level. Some characteristics of threat agents and attack methods can be identified independently, e.g., an agent's motivation and experience. Still, they can also be used as indicators to estimate the likelihood of a threat.

Risk treatment-related concepts describe concepts to treat risk. A *risk treatment-decision* is a decision to treat an identified risk. A treatment satisfies a security need, expressed in generic and functional terms and are refined to security requirements. There are four categories of risk treatment decisions possible – risk avoidance, risk reduction, risk transfer, and risk retention. A *security requirement* is a condition over the phenomena of the environment that we wish to make true by installing the information system, in order to mitigate risks. Finally, a *control* is a designed means to improve the security by implementing security requirements.

Risk treatment and security requirements are estimated in terms of *Risk reduction* performed and *Cost* incurred; Controls – in terms of *Cost*.

Process. In [12], the security risk management process is reported as a analysis result of the security- and security risk management standards. The process in Fig. 3 begins with (*i*) a study of the organization's context and the identification of its assets. Then, one needs to determine the (*ii*) security objectives in terms of confidentiality, integrity and availability of the business assets. The next step of the process is (*iii*) risk analysis where security risks are elicited and assessed. Once risk assessment is finished, decisions about (*iv*) risk treatment are taken. Next step (*v*) is elicitation of security requirements to mitigate the identified risks. Finally, security requirements are implemented to security controls (*vi*). The ISSRM process is iterative. Several iterations need to be performed, until reaching an acceptable level for each risks.

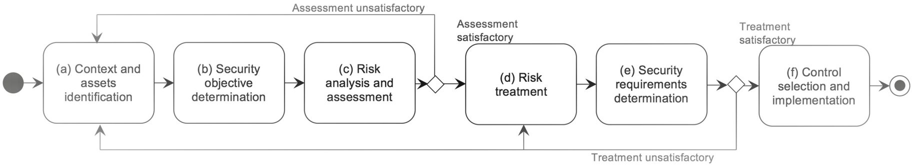

Assessment unsatisfactory
Assessment satisfactory
Treatment satisfactory
(a) Context and assets identification
(b) Security objective determination
(c) Risk analysis and assessment
(d) Risk treatment
(e) Security requirements determination
(f) Control selection and implementation
Treatment unsatisfactory

Fig. 3. Process for security risk management, adapted from [6,12].

3 Asset Identification and Security Objective Determination

In order to analyze security threats in enterprise collaborations, the first step is to identify assets involved in the airline turnaround collaboration of Fig. 1 and determine their security objectives. These two activities correspond to the two first steps illustrated in Fig. 2. We aim to identify assets that are involved in the collaboration between the airline and service providers. Consequently, Sect. 3.1 focuses on the airline turnaround day of operation (DOO). We analyze what IT systems are involved in this turnaround process. Section 3.2 identifies business assets that contain information exchanges between the airline and service providers.

3.1 IT Systems for Airline Turnaround

The airline day of operation shows routine tasks and processes before and after takeoff flights. Following [14], we consider the processes for flight preparation, turnaround and takeoff. Following the depiction in Fig. 1, the process of flight preparation involves gathering and compiling of all flight plans. The latter are documents that describes proposed aircraft flights.

The turnaround phase of operations involves the following set of activities: ground operations, passenger management and gate-agent activities. The ground operations encompass all activities that take place before the passengers start boarding the aircraft. Cargo and luggage offload, aircraft cleaning, restocking of aircraft, re-fueling and loading of cargo and luggage. Passenger management comprises passenger check and luggage check-in activities. The gate agent monitors the ground operations activities and passenger-management activities.

The takeoff activities are the last set of pre-flight activities to be carried out before the actual takeoff of a flight. The activities include reviewing of flight plans, load balancing and the calculation of additionally required fuel. This phase ends with the approval of a flight plan and the request for takeoff clearance from the air-traffic control (ATC).

Out of all the airline DOO processes described, the turnaround phase provides more opportunities for collaborations between the airline and service providers. This is because the activities involved in this phase are resource intensive and are not part of the core competence of the airlines.

From Fig. 1 we derive the following information systems that support collaboration activities in the turnaround process. Passenger management as an IS asset that contains activities, participants, business entities, roles and rules in a passenger-management pool of the turnaround process. Ground operations comprising activities, business entities etc. in the ground operations pool of the turnaround process. The messaging system with rules, protocols and networks that determine how digital information is transmitted between airlines and service providers. For example, the domain-name servers (DNS) and simple mail transfer protocol (SMTP) are networks and protocols that play a role in the delivery of messages from sender to receiver. The passenger check-in process is an IS asset that contains rules, procedures etc. for boarding passengers. Finally, the luggage check-in process as an IS asset that contains rules, procedures etc. on how luggage is checked-in to an aircraft.

3.2 Information Exchange in Collaborating Systems

The knowledge model for airline turnaround is depicted in Fig. 1. It comprises of the following roles - passenger management and ground operations. Each set of activities generates a specific data objects and can also trigger other sets of activities that we give below.

Passenger Management Process: Different forms of passenger data are generated throughout the passenger-management process. Such data may include names, addresses, phone numbers, next of kin, etc. Other data contained may include frequency of travel, destination and hotel reservations of travelers. The value of this information is significant to the airlines because it contains details that completely describe the customers of the airline. Customers (information) are intangible business assets that must be valued and managed [1], and therefore there is a need to secure customer information as a business asset.

The asset identification starts by identifying information-system assets that supports business assets in the passenger-management processes of the airline-turnaround domain. A process description outlining possible activities involving the business asset is shown in the asset identification Tables 1–4 together with the security criteria for each of the business assets identified in the passenger-management processes. Passenger information is contained in the following data objects.

The checked-in passenger information of Table 1 comprises data objects that are generated during the check-in activity and contains data about passengers that checked in for the flight. The data may include time of check-in, seat reservations, and other special requests by passengers. The passenger information also contains personal details such as name, passport number, address, contact, next of kin etc. An attacker with access to this information can successfully conduct social-engineering- or phishing attacks on airline passengers.

The check-in activity represents the IS asset that supports the business asset Checked-in passenger information. An attacker can manipulate the passenger

Table 1. Checked-in passenger information asset identification

Business Asset	Checked-in passenger information	
IS asset	Passenger check-in process; Passenger management; Passenger; Check-in Personnel	
Process description: how do IS assets ***support*** business asset(s)	Passenger *physical check-in* process description: • Passenger goes to the check-in personnel • Passenger provides identification document; • Personnel verifies provided document; • Personnel prints out boarding-pass • Passenger collects boarding pass	Passenger *online check-in* process description: • Passenger visits the online check-in portal; • Passenger enters booking number and confirms check-in; • Passenger prints boarding-pass
Security criteria	Confidentiality of checked-in passenger data	

check-in process and this may cause blacklisted individuals to be able to board the aircraft.

Table 1 shows details about information system assets that support the business asset of checked-in passenger information. The table also shows two process descriptions involving the business assets, namely physical check-in process and online passenger check-in process. The security criterion for the business asset is also identified as confidentiality of checked-in passenger information. Confidentiality of information is necessary because it is important that data contained in checked-in passenger information are only available for people who should have access to it.

The luggage information of Table 2 shows the data object is created in the luggage check-in activity that starts at the end of passenger check-in activity. The luggage information is transmitted via a messaging system to the ground services to generate cargo assignment information. The data object contains details about baggage carried by different passengers. These may include size, weight and content of passenger baggage.

Table 2. Luggage-information asset identification

Business Asset	Luggage information
IS asset	Luggage check-in process; Messaging system; Passenger management
Process description: how do IS assets ***support*** business asset(s)	Luggage check-in process description: • Passenger drops the luggage after passenger check-in process; • Personnel measures luggage to confirm if it meets requirement; • Personnel records the weight and size; • Personnel drops the luggage for loading into aircraft
Security criteria	Confidentiality of luggage information; Integrity of luggage information

The check-in activity and the messaging system represent the IS asset tasks that supports the business asset termed luggage information. An attacker can manipulate the luggage check-in process that may cause luggage with dangerous substances to be loaded to the aircraft.

Table 2 shows details about luggage information business asset. It starts by identifying the IS assets that support the luggage information for the luggage check-in process, the passenger management pool, and the messaging system. The security criteria for the asset are identified as confidentiality of data and

Table 3. Fuel-slip asset identification

Business Asset	Fuel slip
IS asset	Messaging system; Ground operations
Process description: how do IS assets ***support*** business asset(s)	Sending fuel slip to service provider process description: • Service provider receives fuel slip; • Service provider re-fuelling based on data in fuel slip
Security criteria	Integrity of fuel slip

integrity of data. This is important because it is necessary that the data contained in luggage information are only available to the right persons and also remain unchanged.

Ground Operations. The following assets are involved in ground operation activities.

The fuel-slip asset is shown in Table 3. After passengers completely de-board the aircraft, a fuel slip is sent via a messaging system to an external provider to start the refueling activity. The fuel slip contains details about the quantity and quality of fuel to be loaded in various fuel tankers of the aircraft. The messaging system represents an IS asset that supports the fuel-slip-business asset.

The quantity of fuel and distribution in the aircraft is very crucial for evenly spreading weight across the aircraft and maintaining proper load balancing. Controlling the latter refers to the location of the center of gravity of an aircraft. This is of primary importance to aircraft stability and determines safety in flight [1].

The data contained in the fuel receipt can be maliciously changed by an attacker, e.g., to cause not enough fuel to be loaded on the aircraft. It is also possible that an attacker can change the type and quality of fuel on the fuel receipt and this can cause the aircraft to be loaded with wrong fuel. Furthermore, if the wrong quantities of fuel are loaded on different fuel tankers of the aircraft, it can cause the center of gravity of the aircraft to shift beyond allowable limits. This can cause the aircraft to lose stability and spin in midair [1]. Loading an aircraft with the wrong type of fuel results in failure of the engines of the aircraft and can result in air crashes [2].

The fuel slip asset identification in Table 3 shows details of IS assets that support the fuel slip, process description that involves the fuel slip and security criterion for the fuel slip. Two IS assets supporting the fuel slip business asset are messaging system and ground operations pool. The security criterion is identified as integrity of data. It is necessary that data contained in the fuel slip document remains unchanged in the course of airline turnaround.

The cargo assignment in Table 4, commences upon the completion of offloading cargo and luggage. Cargo assignment information is sent via a messaging system to the external provider for commencing the loading of new cargo and luggage into the aircraft. The cargo assignment holds data about weight of baggage, luggage and other check-in cargos. The cargo weights as well as passengers and fuel weights are necessary in maintaining the center of gravity and stability of the aircraft. The messaging system represents the IS asset that supports the business asset cargo assignment.

Table 4. Cargo-assignment asset identification

Business Asset	Cargo assignment
IS asset	Messaging system; Ground operations
Process description: how do IS assets ***support*** business asset(s)	Sending cargo assignment to service provider process description: Service provider receives cargo assignment document; Aircraft is loaded with cargoes and luggage based in data contained in cargo assignment document
Security criteria	Integrity of cargo assignment

An attacker can change the values of data contained in the cargo assignment and cause the aircraft to be overloaded beyond an acceptable weight level. This can reduce the efficiency of the aircraft and also reduce the safety margin available if an emergency condition should arise [1]. The reduction in efficiency of the aircraft can result in the following - higher takeoff speed, longer takeoff run, reduced rate and angle of climb, lower maximum altitude, shorter range, reduced cruising speed, reduced maneuverability, higher stalling speed, higher landing speed, longer landing roll[2].

Table 4 shows the IS assets that support the business asset, process description involving the business asset and security criterion for the business asset. The IS assets are messaging system and ground operation pool while the security criterion is integrity of cargo assignment.

4 Security Risk Analysis

We now apply the ISSRM domain model (see, of Fig. 2) to identify security risks in the *passenger management* and *ground operation* processes. This activity corresponds to the third step of the process illustrated in Fig. 3. The activity starts with identifying potential threat agents, their motivation and the resources that they possess to conduct the attack method. Next, we describe a process about how a threat agent is able to carry out an attack method. The risk analysis, then, continues with the identification of the vulnerability (as a characteristic of the IS asset) and impact that describes how security event harms both business and IS asset and how it negates the security criteria. The risk components (such as threat, event, and risk) are defined as the aggregation of the threat agent, attack method, vulnerability and impact, as described in Fig. 2.

[2] Annex of Weighing-systems.com, describing deficiencies of aircrafts as a result of too much weight, http://tinyurl.com/7kborcf.

In this section, firstly, we detail two risks identified for the *Checked-in passenger information* asset as is defined in Table 1. Next, we overview other security risks pertaining to *luggage information*, *fuel slip*, and *cargo assignment* business assets.

4.1 Analysis of Risks to the *Checked-in Passenger Information* Asset

Two possible attack methods are described for the checked-in passenger information business asset. Each respective attack method has it owns threat agent. Table 5 describes each threat agent, the attack method and risk components for the check-in passenger information.

The analysis in Table 5 starts by identifying two potential threat agents, namely a blacklisted passenger and an attacker. We assume for the blacklisted passenger, his motivation to carry out an attack on check-in passenger information is the need to board the flight that he is blacklisted for. For a random attacker, his motivation is to sabotage the reputation of the airline by causing the passengers to miss their flights. The two possible attackers have knowledge of how the airline check-in process works. We also assume the attacker who wants to harm the reputation of the airline also needs a fake website to carry out a phishing attack [3] on the passengers of the airline.

Table 5 outlines step-by-step details on how these two attacks are carried out against checked-in passenger information. The first attack method describes a physical passenger check-in process while the second attack description shows an attack on online check-in. The vulnerability in the online check-in process is the passenger ignorance of the genuine check-in website. For the physical check-in process, we assume the check-in personnel is bribed. By successfully taking advantage of these weaknesses, an attacker boards a flight with specific passenger information, or even causes a passenger to miss the flight.

The two attacks in Table 5 harm the airline check-in process. The attacks cause the passengers to loose trust in the passenger check-in process and also the passengers' data are stolen in the course of the attack. Specifically for the second attack, the passenger misses the flight as a result of the attack.

4.2 Other Security Risks to Turnaround Assets

Similarly as for the *Check-in passenger information* business asset, we identify other six security risks to the *luggage information*, *fuel slip*, and *cargo assignment* business assets. These risks are:

- **Risk 3**: The personnel records values lower than actual weight of luggage and ground operations uses the information in the loading of the aircraft because of luggage information that results in a loss of integrity of luggage information and an overloading of the aircraft.
- **Risk 4**: The personnel accepts luggage and adds contraband items to a passenger's luggage, sends the luggage and luggage information to ground operations

Table 5. Checked-in passenger information risk- and threat analysis

	Risk 1	Risk 2
Threat agent	*Blacklisted passenger* Motivation: need to board the flight Resources: fake ID, money to bribe the check-in personnel Expertise: knowledge of the check-in process	An attacker Motivations: need to board the flight, sabotage the reputation of the airline and cause airline passengers to miss their flights Resources: fake check-in website, passengers data Expertise: knowledge of check-in process, knowledge email phishing attacks
Attack method	• Bribes personnel to steal checked-in passenger information • Presents fake ID at the check-in desk • Gets checked in with fake ID and checked-in passenger information	• Attacker sends phishing email to passengers that booked a flight • Passenger enters booking number to the fake check-in website and checks in • Passenger prints a fake boarding-pass with flight time changed to few hours ahead of the actual flight time • Attacker uses passenger booking number to check-in to the original site and prints boarding-pass • Attacker boards the flight with the original boarding pass
Threat	Blacklisted passenger bribes the personnel, presents fake ID, and gets checked-in	Attacker uses phishing email to extract passenger-booking number and uses it to check-in to the flight
Vulnerability	Check-in personnel could be bribed	Passenger can't differentiate between original and fake check-in website
Event	Blacklisted passenger presents fake document, bribes personnel and gets checked-in because check-in personnel could be bribed	Attacker uses phishing email to extract passenger booking number and uses it to check-in to the flight because passenger can't differentiate between original and fake check-in website
Impact	Loss of confidentiality of checked-in passenger information; Passenger check-in process can no longer be trusted; Checked- in passenger information is stolen	Loss of trust in online check-in process; Passenger information is stolen; Passenger misses flight
Risk	Blacklisted passenger presents fake document, gets checked-in because personnel could be bribed which results to loss of confidentiality of checked-in passenger, loss of trust in check-in process and stolen checked-in passenger information	Attacker uses phishing email to extract passenger booking number and uses it to check-in to the flight because passenger cant differentiate between original and fake check-in website which causes the passengers to miss their flight, their information stolen, resulting to loss of trust in the airline and its online check-in process

to load the aircraft because personnel activity is not monitored. This results in a loss of integrity of contents pertaining to passenger luggage and a loss of trust in the luggage check-in process.

- **Risk 5**: A malicious insider with access to the computer that stores the fuel slip performs changes to the data contained in the fuel slip before it is sent to a service provider. As the document is not encrypted, it results in a loss of integrity of the fuel slip and causes the aircraft to be loaded with the wrong quantity and type of fuel.
- **Risk 6**: The attacker intercepts the fuel slip, changes the data contained and sends it to the supplier. The re-fuelling is conducted in accordance with information on the fuel slip. As the messaging system is spoofed, it causes a loss of integrity of the fuel slip and results in loading the aircraft with the wrong quantity and type of fuel.
- **Risk 7**: A malicious insider with access rights performs changes to the cargo-assignment document before it is sent to a service provider. As the cargo-assignment document is not encrypted, it causes a loss of integrity of the cargo assignment and an improper loading of the aircraft, leading to instability of the aircraft in the air.
- **Risk 8**: An attacker hacks the airline mailing list, receives the cargo assignment, changes the data contained and sends the cargo assignment to a service provider. The loading is conducted based on the information in the cargo assignment. As the mailing list is not fully secured, it causes a loss of integrity of the cargo assignment and results in overloading the aircraft.

Security risks 3 and 4 are defined towards the *luggage information* business asset. Security risks 5 and 6 are expressed towards the *flue slip* business asset. Finally, Risk 7 and Risk 8 are defined towards the *cargo assignment* business asset.

5 Security Risk Mitigation

In this section, firstly, we cover the three last steps presented in Fig. 3. In all cases, to mitigate the identified security risks, the *risk reduction* treatment decision is taken. The latter we refine to security requirements for implementation with existing security controls.

Security requirements and controls suggestions to mitigate the identified risks we provide in Table 6. The security requirement needed to reduce *Risk 1* is monitoring the activities of check-in personnel. In order to achieve this security requirement, a control is applied. The security control requires an additional officer to always verify the activities for the check-in personnel. The cost value of 4 implies that it costs more to employ additional staff to verify the activities of check-in personnel.

For *Risk 2*, the security requirement to reduce the risk is achieved by educating airline passengers on possible phishing-attack methods. To achieve this, a security control is applied by using only secured https websites for booking and passenger check-in. The cost value of 2 implies that it costs less to educate the

Table 6. Security requirements and counter-measures

Risk	Security requirement	Control	
		Description	Cost
Risk 1	Monitor the activity of check-in personnel	Officer verifying the actions of check-in personnel	4
Risk 2	Educate the airline passengers on phishing attacks	Using secured https websites for booking and check-in activity	2
Risk 3	Monitor activities of luggage check-in personnel	Random checks to verify weight records with actual weight of luggage	2
Risk 4	Monitor activities of luggage check-in personnel	Install camera to record luggage check-in personnel activities and scanning the recordings	2
Risk 5	Access control on fuel slip document	Encrypt fuel slip document	4
Risk 6	Make information contained in fuel slip unreadable	Verifies received document with previous originals received (Blockchain cryptographic digest PKI or PGP) Encrypt fuel slip document	4
Risk 7	Access control on cargo assignment document	Encrypt cargo assignment document	4
Risk 8	Make information contained in cargo assignment unreadable	Verify received document with previous originals received. Encrypt cargo assignment document	4

passengers against phishing methods and provide secured website for booking and check-in activities.

The security requirement for reducing *Risk 3* is monitoring activities of luggage check-in personnel. The security control that must be implemented to achieve this is performing random checks on activities of luggage check-in personnel. The risk treatment value of 2 implies that less cost is required to perform checks that verify data recorded by personnel.

For *Risk 4*, the security requirement is the same as for Risk 3 which is monitoring activities of personnel. However, a different security control is applied. The control is achieved by installing cameras to record activities of luggage check-in personnel and scanning the recordings. The treatment cost of 1 shows that little money is required to mount security cameras to monitor activities of check-in personnel.

The security requirement for *Risk 5* is controlling access to the fuel slip document. Access control for the fuel slip is achieved by encrypting the fuel slip document. Such encryption that relies on a public-key infrastructure (PKI) is applicable in this case. The fuel slips and other documents that contain service requirements are encrypted with private keys of the selected supplier and therefore, only the supplier views the document, even when intercepted by another person.

For *Risk 6*, the same security requirement and security control apply as in *Risk 5*. The risk treatment value of 4 for both risks implies that it costs the airline a lot to implement an encryption that uses PKI.

The security requirements and controls for *Risk 7* and *Risk 8* are the same. In order to reduce these risks for the cargo-assignment document, proper access control must be implemented. The latter is achieved by applying encryption that relies on PKI. As a result, only the service provider can have access to the document, even when interception occurs. The cost value of 4 implies that it is expensive to implement encryption that depends on PKI in order to reduce *Risk 7* and *Risk 8*.

6 Risk Assessment and Control Selection

In this section we return to Step 3 with a focus on *security risk assessment* and Step 6 with a focus on *control selection* presented in Fig. 3. Hence we assess how the security requirements and controls of Sect. 5 affect the identified security risks of Sect. 4. For that assessment, we use the goal question metric (GQM) [21]. The ISSRM approach [12] suggests a number of questions to maximize risk reduction and minimize risk-treatment costs. In this context, our emphasis is placed on risk-reduction estimation.

6.1 Security Risk Assessment

The GQM-questions target the risk level, its occurrence frequency, importance regarding the business, the risk-reduction level after treatment of risk. The risk management metrics, such as business asset value, threat likelihood, vulnerability level, and security objective are estimated in the scale from 0 (lowest) to 5 (highest). The risk event, risk impact and risk level are then calculated as follows:

- Risk event = threat likelihood + vulnerability level -1
- Impact = maximum value of the security criterion
- Risk level = risk event x impact.
- Maximum-risk level = $(5 + 5 - 1) * 5 = 45$
- Minimum-risk level = $(0 + 0 - 1) * 0 = 0$
- Risk reduction level = Risk level 1 − Risk level 2

The minimum-risk level obtainable is 0, while the maximum-risk level obtainable is 45. Therefore, 0 and 45 represent the boundaries of the risks. Here, Risk level 1 is calculated when no security countermeasures are applied and Risk level 2 is calculated after the application of the security countermeasures. The collected data are illustrated in Table 7.

Table 7. Risk metrics before and after risk treatment.

	Before treatment					After treatment				Risk reduction level	Business asset value	Cost of counter-measure
	Vulnerability level	Threat likelihood	Event potentiality	Impact level	Risk level1	Vulnerability level	Threat likelihood	Event potentiality	Risk level2			
Risk1	3	2	4	3	12	2	1	2	6	6	3	4
Risk2	2	4	5	3	15	1	3	3	9	6	3	2
Risk3	1	2	2	2	4	1	1	1	2	2	1	2
Risk4	4	2	5	3	15	2	1	2	6	9	1	1
Risk5	3	3	5	4	20	1	1	1	4	16	3	4
Risk6	3	2	4	4	16	1	1	1	4	12	3	4
Risk7	2	3	4	3	12	1	1	1	4	8	1	4
Risk8	2	2	3	3	9	1	2	2	6	3	1	4

6.2 Security Control Selection

Not all security risks of our study can be mitigated, e.g., because of lacking resources, or time-to-market requirements. Thus, security control selection, potentially, means understanding which controls need to be selected, or in other words which security risks need to be mitigated first. One way to perform this trade-off analysis is by using the *value* of the business asset, *counter-measure cost* and *risk reduction level* (RRL), gathered in Table 7 (see, three last columns). Using these metrics, three graphs are prepared (see Figs. 4, 5 and 6) including data on RRL and value, RRL and cost, and cost and value. The graphs are divided into four quadrants and the priority on each quadrant is identified by labels low (L), medium (M) and high (H) on each quadrant.

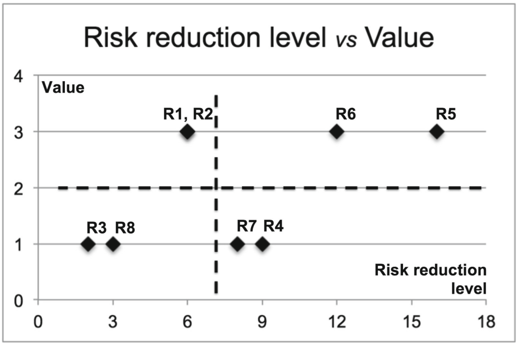


Fig. 4. Risk-reduction level against business asset value.

Figure 4 shows a graph about the risk-reduction level against business asset value. The desired situation is a high value asset with a high risk reduction value. This can be identified in the quadrant that has Risk 6 (R6), R5 and therefore represents a high priority. The medium priorities quadrants have high asset value with low risk-reduction level and low-value assets coupled with a high risk-reduction values. These situations are found in quadrants that have R1, R2 and R7, R4 respectively. The least desired situation is a low valued asset with a low risk reduction in quadrant that has R3 and R8.

Figure 5 shows a graph of risk-reduction level against cost of counter measure. The ideal situation is a low cost value with a high risk reduction value. This can be identified in the quadrant comprising R4 and therefore, it represents a high priority. The medium-priority quadrants have high cost value with high risk-reduction level and low cost with low risk-reduction values. These situations are found in quadrants that comprise R5, R6, R7 and R2, R3 respectively. The low priority can be identified in the quadrant of high cost and low risk reduction. This quadrant contains R1 and R8.

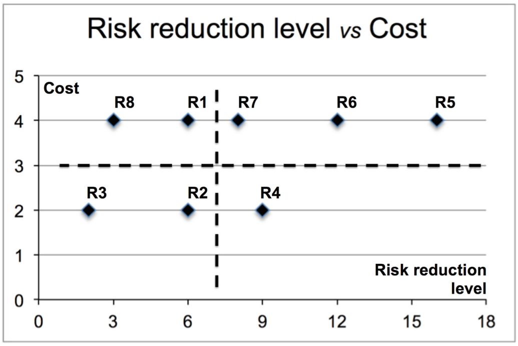


Fig. 5. Risk-reduction level against cost of counter measure.

Figure 6 is about the cost of counter measure against business asset value. A low cost treatment with a high-value asset represents a high priority and can be seen in the quadrant with R2. The medium priority are found in quadrants combining high-value assets with high cost of counter measure and low-value assets and low cost of counter measure. These are found in the quadrant comprising R1, R5, R6 and the quadrant with R3, R4. The least ideal situation is a low-value asset with a high cost of risk treatment and it is found in the quadrant with R7, R8.

Table 8 shows risk priorities derived from combining the graphs of Figs. 4, 5 and 6 where a value of 1 is assigned to low priority risks, medium priority risks has a value of 2, while the value of 3 is assigned to high priority risks. By adding these values across the three graphs, a priority can be estimated that depends on the value of a business asset, cost of counter measure and risk reduction level. The risks with high priorities are R2, R4, R5 and R6. The medium priority risks are R1, R3 and R7. The least priority risk is R8.

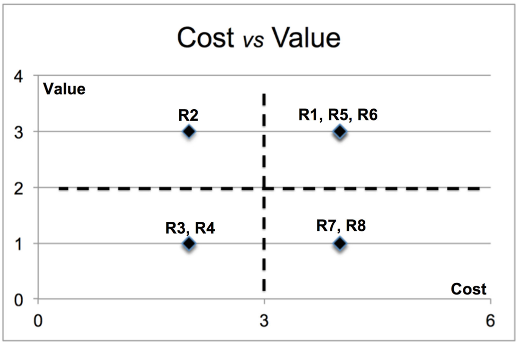
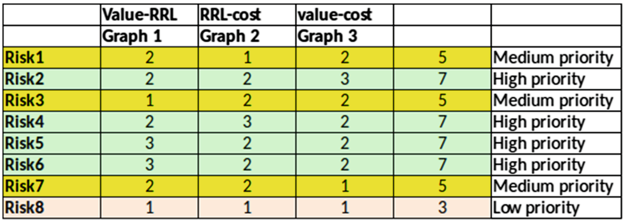


Fig. 6. Cost of counter measure against business asset value.

Table 8. Risk versus priority



	Value-RRL	RRL-cost	value-cost		
	Graph 1	Graph 2	Graph 3		
Risk1	2	1	2	5	Medium priority
Risk2	2	2	3	7	High priority
Risk3	1	2	2	5	Medium priority
Risk4	2	3	2	7	High priority
Risk5	3	2	2	7	High priority
Risk6	3	2	2	7	High priority
Risk7	2	2	1	5	Medium priority
Risk8	1	1	1	3	Low priority

7 Conclusions

In this paper, security issues affecting cross-organizational collaborations are analyzed using the aviation sector. The airline turnaround process presents a good environment for this study as operations are resource intensive. The end result of the analysis in this work is a set of security requirement- and controls for managing risks resulting from the collaboration between airlines and service providers.

The relevant assets for collaborations between enterprises are identified by describing airline turnaround processes. We identify the IS assets in the passenger-management process and ground operations. The risks identified in the risk analysis section are reduced by applying security requirement and controls for each risk identified. We find the security requirements for the reduction of risks identified for the checked-in passenger information, the luggage information, the security requirements. Likewise, we detect the security requirements for the reduction of risks identified in the ground operations for the fuel slip and the cargo assignment.

The security controls for the reduction of risks are listed as follows: For the checked-in passenger information, the security controls are an officer verifying the actions of check-in personnel and using only secured https websites for booking and check-in activity. For the luggage information, the security controls are random checks to verify weight records with actual weight of luggage and installing cameras to record luggage check-in personnel activities. The security controls for the reduction of risks identified in ground operations are listed as follows. For the fuel slip, the security controls are encrypting fuel-slip documents, and verifying all received documents and comparing them with previously received originals. For the cargo assignment, the security controls are encrypt the cargo-assignment document, and verify all received documents and compare with previous originals received. Furthermore, we show the degree of security that is achieved with implementing security controls.

As future research we plan to apply risk-based patterns in modeling a secure business process for cross-organizational collaborations and also intend to analyze security threats in cloud-supported enterprise collaborations. Note that a corresponding research paper [18] is forthcoming. Additionally, we intend to further explore methods of data collection for improved risk-metrics calculations.

References

1. US Department of Transportation: Aircraft weight and balance handbook (2007). http://tiny.cc/m7xkcy
2. NATA Safety 1st eToolkit (2015). http://tiny.cc/5nzkcy
3. Anton, V.U., Eduardo, B.F.: An extensible pattern-based library, taxonomy of security threats for distributed systems. Secur. Inf. Syst. Adv. New Challenges **36**, 734–747 (2014)
4. Bartelt, C., Rausch, A., Rehfeldt, K.: Quo vadis cyber-physical systems: research areas of cyber-physical ecosystems: a position paper. In: Proceedings of the 1st International Workshop on Control Theory for Software Engineering, CTSE 2015, pp. 22–25. ACM, New York (2015)
5. Belobaba, P., Odoni, A., Barnhart, C.: The global airline industry. Wiley, Chichester (2015)
6. Dubois, E., Heymans, P., Mayer, N., Matulevičius, R.: A systematic approach to define the domain of information system security risk management. In: Nurcan, S., Salinesi, C., Souveyet, C., Ralyté, J. (eds.) Intentional Perspectives on Information Systems Engineering, pp. 289–306. Springer, Heidelberg (2010)
7. Kutvonen, L., Norta, A., Ruohomaa, S.: Inter-enterprise business transaction management in open service ecosystems. In: 2012 IEEE 16th International on Enterprise Distributed Object Computing Conference (EDOC), pp. 31–40. IEEE (2012)
8. Leonardi, M., Piracci, E., Galati, G.: Ads-b vulnerability to low cost jammers: risk assessment and possible solutions. In: 2014 Tyrrhenian International Workshop on Digital Communications-Enhanced Surveillance of Aircraft and Vehicles (TIWDC/ESAV), pp. 41–46. IEEE (2014)
9. Long, S.: Socioanalytic Methods: Discovering the Hidden in Organisations and Social Systems. Karnac Books, London (2013)
10. Maiden, Neil Arthur McDougall, Ncube, Cornelius, Lockerbie, James: Inventing Requirements: Experiences with an Airport Operations System. In: Rolland, Colette (ed.) REFSQ 2008. LNCS, vol. 5025, pp. 58–72. Springer, Heidelberg (2008)
11. Massacci, F., Paci, F., Tedeschi, A.: Assessing a requirements evolution approach: empirical studies in the air traffic management domain. J. Syst. Soft. **95**, 70–88 (2014)
12. Mayer, N.: Model-based management of information system security risk. Ph.D. thesis. University of Namur (2009)
13. Business Process Model. Notation (bpmn) version 2.0. Object Management Group specification (2011). http://www.bpmn.org
14. Nõukas, R.: Service brokering environment for an airline, (Master Thesis). Tallinn University of Technology (2015)
15. Norta, Alex: Creation of Smart-Contracting Collaborations for Decentralized Autonomous Organizations. In: Matulevičius, Raimundas, Dumas, Marlon (eds.) BIR 2015. LNBIP, vol. 229, pp. 3–17. Springer, Heidelberg (2015)
16. Norta, A., Grefen, P., Narendra, N.C.: A reference architecture for managing dynamic inter-organizational business processes. Data Knowl. Eng. **91**, 52–89 (2014)
17. Norta, A., Ma, L., Duan, Y., Rull, A., Kõlvart, M., Taveter, K.: eContractual choreography-language properties towards cross-organizational business collaboration. J. Internet Serv. Appl. **6**(1), 1–23 (2015)
18. Samarütel, S., Matulevičius, R., Norta, A., Nõukas, R. In: Horkoff, J., Jeusfeld, M., Persson, A. (eds.) The Practice of Enterprise Modeling. LNBIP, vol. 267, 1st edn. Springer, Heidelberg (2016)

19. Sampigethaya, K., Poovendran, R.: Aviation cyber-physical systems: foundations for future aircraft and air transport. Proc. IEEE **101**(8), 1834–1855 (2013)
20. Shim, W., Massacci, F., Tedeschi, A., Pollini, A.: A relative cost-benefit approach for evaluating alternative airport security policies. In: 2014 Ninth International Conference on Availability, Reliability and Security (ARES), pp. 514–522. IEEE (2014)
21. van Solingen, R., Basili, V., Caldiera, G., Rombach, H.D.: Goal Question Metric (GQM) Approach. Wiley, New York (2002)

Collective Anomaly Detection Based on Long Short-Term Memory Recurrent Neural Networks

Loïc Bontemps, Van Loi Cao(✉), James McDermott, and Nhien-An Le-Khac

University College Dublin, Dublin, Ireland
{loic.bontemps,loi.cao}@ucdconnect.ie,
{james.mcdermott2,an.lekhac}@ucd.ie

Abstract. Intrusion detection for computer network systems is becoming one of the most critical tasks for network administrators today. It has an important role for organizations, governments and our society due to the valuable resources hosted on computer networks. Traditional misuse detection strategies are unable to detect new and unknown intrusion types. In contrast anomaly detection in network security aims to distinguish between illegal or malicious events and normal behavior of network systems. Anomaly detection can be considered as a classification problem where it builds models of normal network behavior, which it uses to detect new patterns that significantly deviate from the model. Most of the current research on anomaly detection is based on the learning of normal and anomaly behaviors. They have no memory that is they do not take into account previous events classify new ones. In this paper, we propose a real time collective anomaly detection model based on neural network learning. Normally a Long Short-Term Memory Recurrent Neural Network (LSTM RNN) is trained only on normal data and it is capable of predicting several time steps ahead of an input. In our approach, a LSTM RNN is trained with normal time series data before performing a live prediction for each time step. Instead of considering each time step separately, the observation of prediction errors from a certain number of time steps is now proposed as a new idea for detecting collective anomalies. The prediction errors from a number of the latest time steps above a threshold will indicate a collective anomaly. The model is built on a time series version of the KDD 1999 dataset. The experiments demonstrate that it is possible to offer reliable and efficient collective anomaly detection.

Keywords: Long short-term memory · Recurrent neural network · Collective anomaly detection

1 Introduction

Network anomaly detection refers to the problem of detecting illegal or malicious activities or events from normal connections or expected behavior of network systems [4,5]. It has become one of the most popular subjects in the network

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 141–152, 2016.
DOI: 10.1007/978-3-319-48057-2_9

security domain due to the fact that organizations and governments are now seeking good solutions to protect valuable resources on computer networks from unauthorized and illegal accesses, network attacks or malware. Over the last three decades, machine learning techniques are known as a common approach for developing network anomaly detection models [3,4]. Network anomaly detection is usually posed as a type of classification problem: given a dataset representing normal and anomalous examples, the goal is to build a learning classifier which is capable of signaling when a new anomalous data sample is encountered [5].

However, most of the existing approaches consider an anomaly as a single point: cases when they occur "individually" and "separately" [6,7,16]. In such approaches, anomaly detection models do not have the ability to represent the information from previous data or events for evaluating a current point. In network security, some kinds of attacks, *Denial of Service (DoS)*, usually occur for a long period of time (several minutes) [10], and are often represented by a set of single points. An attack should be indicated only if a set of single points are considered as an attack. In order to detect this kind of attack, anomaly detection models should be capable of remembering the information from a number of previous events, and representing the relationship between them and the current event. To avoid important mistakes, one must always consider every outcome: in this sense a highly anomalous value may still be linked to a perfectly normal condition, and conversely. In this work, we aim to build an anomaly detection model for this kind of attacks (known as *collective anomaly detection* in [5]).

Collective anomaly is the term to refer to a collection of related anomalous data instances with respect to the whole dataset [5]. The single data points in a collective anomaly may not be considered as anomalies by themselves, but the occurrence of these single points together indicates an anomaly. Long Short-Term Memory Recurrent Neural Network (LSTM RNN) is known as a powerful technique to represent the relationship between a current event and previous events, and handles time series problems [12,14]. Thus, it is employed to develop an anomaly detection model in this paper.

In this paper, we will propose a collective anomaly detection model by using the predictive power of LSTM RNN [8]. Firstly, LSTM RNN is applied as a time series anomaly detection model. The prediction of a current event will depend on both the current event and its previous events. Secondly, the model will be adapted to detect collective anomalies by proposing a circular array. The circular array contains the prediction errors from a certain number of recent time steps. If the prediction errors in the circular array are higher than a predetermined threshold and last for a certain time steps, it will indicate a collective anomaly. More details will be described in Sect. 4.

The rest of the paper is organized as follows. We briefly review some work related to anomaly detection and LSTM RNN. In Sect. 3, we give a short introduction to LSTM RNN. This is followed by a section proposing the collective anomaly detection model using LSTM RNN. Experiments, Results and Discussion are presented in Sects. 5 and 6 respectively. The paper concludes with highlights and future directions.

2 Related Work

When considering a time series dataset, point anomalies are often directly linked to the value of the considered sample. However, attempting real time collective anomaly detection implies always being aware of previous samples, and more precisely their behavior. This means that every time step should include an evaluation of the current value combined with the evaluation of preceding information. In this section, we briefly describe work applying LSTM RNN to time series and collective anomaly detection problems [12,14,15].

Olsson et al. [15] proposed an unsupervised approach for detecting collective anomalies. In order to detect a group of the anomalous examples, the "anomalous score" of the group of data points was probabilistically aggregated from the contribution of each individual example. Obtaining the collective anomalous score was carried out in an unsupervised manner, thus it is suitable for both unsupervised and supervised approaches to scoring individual anomalies. The model was evaluated on an artificial dataset and two industrial datasets, detecting anomalies in moving cranes and anomalies in fuel consumption.

In [12], Malhotra et al. applied a LSTM network for addressing the problem of time series anomaly detection. A stacked LSTM network trained on only normal data was used to predict over a number of time steps. They assumed that the resulting prediction errors have a Gaussian distribution, which was used to assess the likelihood of anomaly behavior. Their model was demonstrated to perform well on four datasets.

Marchi et al. [13,14] presented a novel approach by combining non-linear predictive denoising autoencoders (DA) with LSTM for identifying abnormal acoustic signals. Firstly, LSTM Recurrent DA was employed to predict auditory spectral features of the next short-term frame from its previous frames. The network trained on normal acoustic recorders tends to behave well on normal data, and yields small reconstruction errors whereas the reconstruction errors from abnormal acoustic signals are high. The reconstruction errors of the autoencoder was used as an "anomaly score", and a reconstruction error above a predetermined threshold indicates a novel acoustic event. The model was trained on a public dataset containing in-home sound events, and evaluated on a dataset including new anomaly events. The results demonstrated that their model performed significantly better than existing methods. The idea is also used in a practical acoustic example [13,14], where LSTM RNNs are used to predict short-term frames.

The core idea of this paper is to combine the previous methods, to adapt Long Short-Term Memory to collective anomaly detection. By labelling testing LSTM RNN outputs at every time step with a standardized error value, we shall propose an algorithm to detect collective anomalies. This will prove very useful in our example: First, we will train normal data on an LSTM RNN in order to estimate the behaviour of a normal day of traffic. Then, we will use a classifier inspired by [15] to rate the level of anomaly of each time sample. We will apply this method to a network security problem (KDD 1999 cup), aiming to raise an alarm in the case of DoS Neptune attacks.

3 Preliminaries

In this section, we briefly describe a specific type of Recurrent Neural Network: Long Short Term Memory. The structure was proposed by Hochreiter et al. [8] in 1997, and has already proven to be a powerful technique for addressing the problem of time series prediction.

The difference initiated by LSTM regarding other types of RNN resides in its "smart" nodes presented in Fig. 1. Each of these cells contains three gates, input gate, forget gate and output gate, which decide how to react to an input. Depending on the strength of the information each node receives, it will decide to block it or pass it on. The information is also filtered with the set of weights associated with the cells when it is transferred through these cells.

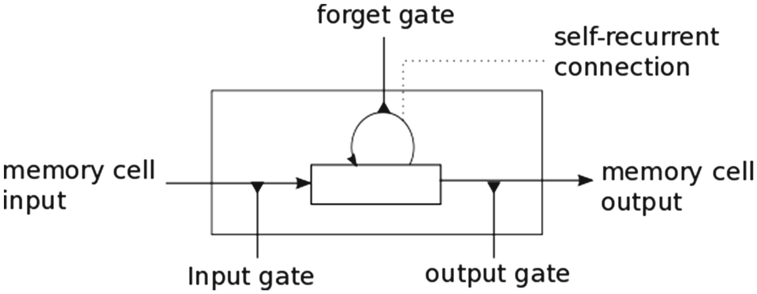


Fig. 1. LSTM RNN Cell, figure reproduced from [1]

The LSTM node structure enables a phenomenon called backpropagation through time. By calculating for each hidden layer the partial derivatives of the output, weight and input values, the system can move backwards to trace the evolving error between real output and predicted output. Afterwards, the network uses the derivative of this evolution to adapt its weights and decrease prediction error. This learning method is named Gradient Descent.

As mentioned before, Long Short-Term Memory has the power to incorporate a behaviour into a network by training it with normal data. The system becomes representative of the variations of the data. In other words, a prediction is made focusing on two features: the value of a sample and its position at a specific time. This means that two input samples at different times may have the same value, but their outputs will very probably differ. It is because a LSTM RNN is stateful, i.e. has a "memory", which changes in response to inputs.

4 Proposed Approach

In this section, we are going to describe a new approach to address the problem of collective anomaly detection. Firstly, we show the LSTM RNNs ability to learn the behaviour of a training set, and in this stage it acts like a time series anomaly detection model. We will then adapt it for collective anomaly detection by introducing terms that measure its prediction errors in a period of time steps. Finally, we shall describe how to seek a collective anomaly by combining a LSTM RNN with a circular array method.

4.1 LSTM RNN as a Predictive Vector

The first step is inspired by the idea presented in [12]: when trained correctly, LSTM RNNs have the ability to learn the behavior of a training set. Intuitively, this means that when given certain input samples, they have the ability to remember the context of the samples, and to predict a coherent output in agreement with that context. In our work, we will use a simple LSTM RNN, in contrast to a stacked LSTM in [12]. This does not change the core principle of the method: when given sufficient training, a LSTM RNN adapts its weights, which become characteristic of the training data.

4.2 Definitions

In order to adapt a LSTM RNN for time series data to detect collective anomalies, we introduce terms to measure prediction errors at each time step or in a period of time steps. These terms are defined as below.

- **Relative Error (RE):** the Relative Error between two real values x and y is given by Eq. 1:

$$RE\left(x, y\right) = \frac{|x - y|}{x} \tag{1}$$

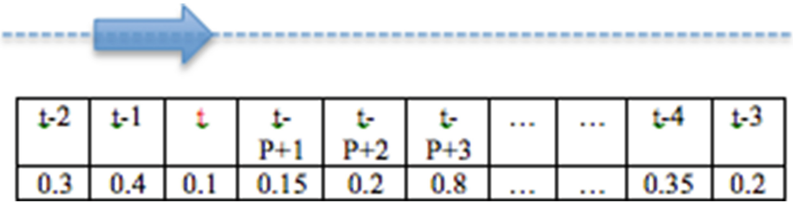
- **Relative Error Threshold (RET):** Relative Error value above a predetermined threshold indicates an anomaly. This threshold, RET, is determined by using labeled normal and attack data from a validation set.
- **Minimum Attack Time (MAT):** The minimum amount of recent time steps that is used to define a collective attack.
- **Danger Coefficient (DC):** The density of anomalous points within the last MAT time steps. Let N be the number of anomalous points over the last MAT time steps, DC is defined as in Eq. 2.

$$DC = \frac{N}{MAT} \tag{2}$$

  NB: $0 < DC < 1$
- **The Averaged Relative Error (ARE):** The Average Relative Error over a MAT is given by Eq. 3:

$$ARE = \sum_{i=1}^{MAT} RE_i \tag{3}$$

The values of two terms, *Danger Coefficient* and *Average Relative Error*, are the key factors that will help the model to decide whether a set of inputs within a number of the latest time steps is a collective anomaly or not as described in Sect. 4.3. These values will be estimated by using a validation set.



t-2	t-1	t	t-P+1	t-P+2	t-P+3	...	...	t-4	t-3
0.3	0.4	0.1	0.15	0.2	0.8	...	...	0.35	0.2

Fig. 2. Circular array for collective anomaly detection model, $MAT = P$

4.3 Degree of Error Evaluation

At each time step, the sample predicted by the LSTM RNN is compared with the real future sample. This comparison is computed as a *RE* value. In this sense, a "Relative Error time series" is built online. Based on the values in a validation set, we can initialise the RET values.

At this stage, our system is theoretically capable of detecting point anomalies at each time step. In order to adapt the model from an individual anomaly model to a collective anomaly one, we must consider simultaneously an ensemble of points. To do this, we propose a circular array containing the MAT latest error values to represent the level of anomaly of the latest time steps as shown in Fig. 2. By analyzing the circular array at every time step, we evaluate the possibility of facing a collective anomaly. A collective anomaly will be identified if both *Danger Coefficient* and *Average Relative Error* are higher than predefined thresholds, α and β, respectively (α and β will be estimated by using the validation set).

5 Experiments

5.1 Datasets

In order to demonstrate the efficient performance of the proposed model, we choose a dataset related to the network security domain, the KDD 1999 dataset [2,9], for our experiments. The dataset in tcpdump format was collected from a simulated military-like environment over a period of 5 weeks. There are four main groups of attacks in the dataset, but we restrict our experiments on a specific attack, *Neptune*, in the Denial-of-Service (DoS) group. The dataset is also converted into a time series version before feeding into the model. More details about how to obtain a time series version from the original dataset, and how to choose training, validation and testing sets are presented in the following paragraphs.

The first crucial step is to build a conveniently usable time series dataset out of the tcpdump data, and to select the features we wish to use. We use terminal commands and a python program to convert the original tcpdump records in the KDD 1999 dataset into a time dependant function. This method is a development of the proposed transformation in [11] that acts directly on the tcpdump to obtain real time statistics of the data. Our scheme follows this step by step transition as described below:

$$\text{tcpdump} \Rightarrow \text{pcap} \Rightarrow \text{csv} \tag{4}$$

Each day of records can be time-filtered and input into a new *.pcap* file. This also has the advantage of giving a first approach on visualizing the data by using Wireshark functionalities (IO graphs and filters). Once this is done, the *tshark* command is adapted to select and transfer the relevant information from the records into a *.csv* file. We may note that doing this is a first step towards faster computation and better system efficiency, since all irrelevant pcap columns can be ignored. There are two major steps for the conversion processing.

1. Store the information of a *.tcpdump* file into a newly generated *.pcap* file. From the terminal, we use the *editcap* command:

```
editcap -A'1999-03-11 08:00:00' -B'1999-03-11 18:00:00'
Thursday2outside.tcpdump Thursday2.pcap
```

2. Convert from *.pcap* file into *.csv* file by *tshark* command.
From the terminal again, type the command below:

```
tshark -r Thursday2.pcap -T fields -e frame.number -e frame.len
-e frame.time -e ip.proto -E header=y -E separator=, -E quote=d
-E occurrence=f -i netstat -f tcp[13]==12  > Thursday2.csv
```

tshark is a simple but powerful command, enabling the selection of columns of interest in a *.pcap* file, and their output in a newly generated *.csv*. Once the data is in the *.csv* format, python code can be implemented from the XX library to store it and use with our classifier.

Processing the tcpdump with this method enables quick and easy manipulation of the data. For example, Neptune and Smurf are both DoS attacks characterised by a high flow of specific packets in networks (eg. SYN_ACK and ICMP echo replies). By using this simple fact, the needed records can be filtered and counted at every time step. If we aim to detect Neptune attack, the *thark* command can be implemented with the -i netstat -f tcp[13] == 2 filter, so only SYN_ACK packets from servers are counted. We observe in the case of KDD 1999 that a Neptune attack can be sought by looking for an anomalously high number of these packets.

The KDD1999 time series is composed of a two-weeks training set n_1 (weeks 1 & 3, normal data), one week of validation set v_1 (week 2, both labeled normal and anomaly data), and a two-week testing set t_1 (weeks 4 & 5). The protocol will be the following: training the network with n_1, using v_1 to determine our error threshold(s), and evaluating the proposed model on t_1.

5.2 Experimental Settings

In this work, we conduct two experiments, one preliminary experiment and one main experiment. The preliminary experiment aim to estimate the parameters for the model and set its thresholds by using the validation set whereas the main experiment is to evaluate the proposed model.

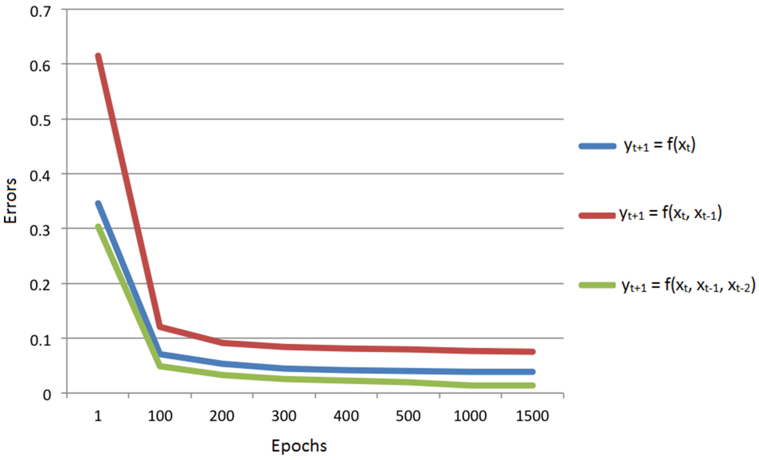


Fig. 3. The training errors from the model with one, two and three inputs

Preliminary Experiment: This experiment aim to select the best parameters of our LSTM RNN model with respect to minimize its prediction error, and determine the thresholds, α and β. Firstly, we determine how many previous time steps should be used for predicting the current event. The hyper-parameters of LSTM RNN, hidden size and learning rate, are then estimated. Finally, the two thresholds, α and β, will be chosen to give the best possible classification performance of the model on the validation set.

In order to optimize the proposed model for the main experiment, we proceed to a preliminary test to measure the influence of the number of inputs on the prediction error of LSTM. We first focus on how many inputs will influence the prediction of an LSTM [12]. We form the hypothesis that inserting more values in our system may help decrease prediction errors, but it will be more time consuming [12]. Thus, we investigate the relationship between the prediction value y_{t+1} to three sets of the previous input examples (x_t), (x_t, x_{t-1}) and (x_t, x_{t-1}, x_{t-2}). They are formulated in Eqs. 5, 6 and 7 below:

$$y_{t+1} = f\left(x_t\right) \tag{5}$$

$$y_{t+1} = f\left(x_t, x_{t-1}\right) \tag{6}$$

$$y_{t+1} = f\left(x_t, x_{t-1}, x_{t-2}\right) \tag{7}$$

where x_t, x_{t-1} and x_{t-2} are the input samples at times t, $t-1$ and $t-2$ respectively, and y_{t+1} is the predicted value for the input x_t.

The number of hidden nodes and the learning rate are the final two parameters that can strongly influence the performance of a LSTM RNN. On the one hand, the strength of a LSTM RNN resides in its hidden layer. Each synapse of a network is weighted differently, and can be considered as a unique interpretation of the input data. Each node of the hidden layer is storage space for these interpretations. Theoretically, the higher number of hidden nodes, the more information the network can contain. This also means more computation, and may lead to over-fitting.

Using the LSTM RNN error evolution curve empirically, we concluded that the optimum number of nodes in our hidden layer to obtain good memorization is approximately *23*, but the results are not shown in this paper. The learning rate is another factor directly linked to the speed at which a LSTM RNN can improve its predictions. For a time step t during training, the synapse weights of our neural network are updated. The learning rate defines how much we wish a weight to be modified at each instant. In our experiment, we choose learning rate equal to *0.01* that gives us a convenient error curve.

Finally, a classifier that is trained on ten days of normal data is used to determine α and β. We observe the reaction of the system on labeled Neptune attacks from the validation set, and set the thresholds. The values of these thresholds is shown in Sect. 6.

Main Experiment: Our task is to use the potential speed and accuracy of LSTM RNN to detect a disproportionate durable change in a time series. Once the preliminary experiment is complete, we choose the most performant LSTM RNN architecture, and train it with the normal training set n_1. The classifier is then evaluated on testing set t_1 containing both normal and attack data to investigate how efficiently our proposed classifier performs.

6 Results and Discussion

This section presents our experimental results. First, the preliminary experiment evaluates two factors: computation cost and LSTM prediction error when using one input, two inputs and three inputs respectively. Then, the general performance in terms of classification accuracy is measured (Fig. 4).

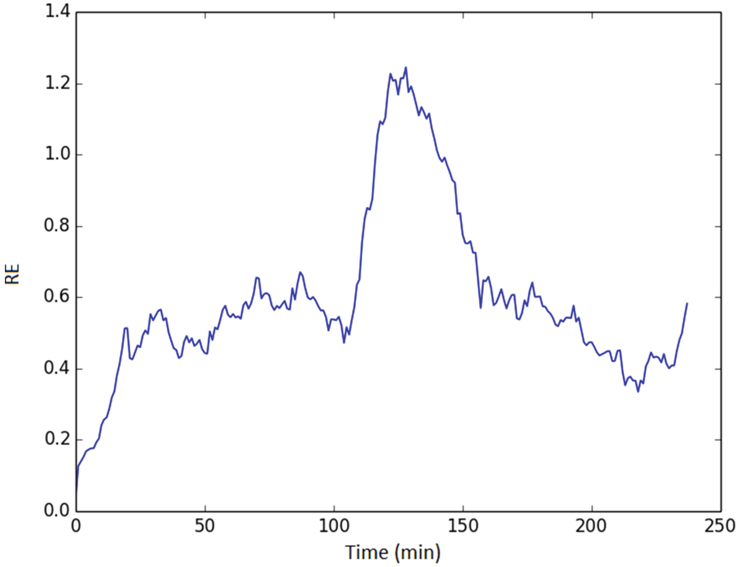


Fig. 4. The prediction error from the model with three inputs (1500 Epochs)

The Table 1 illustrates that the model with three inputs had less computational time than those with one or two inputs. Moreover, the Fig. 3 shows that the model with three inputs achieves a lower training error in comparison to two others. Thus, we use the model with three inputs for our main experiment.

Table 1. Computational time recording

Number of inputs	Computational time (s)
1	645
2	652
3	642

The results from the main experiment are shown in Table 2. The experiment is done with $MAT = 12$, and $\alpha = 0.66$, and we also report the results on four values of β, $\beta = 0.69, 0.66, 0.62$ and 0.52. We observe that it is possible to obtain 100 % collective anomaly detection rate, but this implies triggering a high amount of false alarms. Conversely, it is possible to avoid false alarms, but fewer correct alarms will be detected. Ultimately, detecting more real attacks results in triggering more false alarms as shown in Table 2.

Table 2. Circular array detection efficiency

Threshold β	Percentage of correct alarms triggered	Number of false alarms triggered
0.69	86 %	0
0.66	94 %	2
0.62	98 %	16
0.52	100 %	63

7 Conclusion and Further Work

In this paper, we have proposed a model for collective anomaly detection based on Long Short-Term Memory Recurrent Neural Network. We have motivated this method through investigating LSTM RNN in the problem of time series, and adapted it to detect collective anomalies by proposing the measurements in Sect. 4.2. We investigated the hyper-parameters, the suitable number of inputs and some thresholds by using the validation set.

The proposed model is evaluated by using the time series version of the KDD 1999 dataset. The results suggest that proposed model is efficiently capable of detecting collective anomalies in the dataset. However, they must be used with caution. The training data fed into a network must be organized in a coherent manner to guarantee the stability of the system. In future work, we will focus on how to improve the classification accuracy of the model. We also observed that implementing variations in a LSTM RNNs number of inputs might trigger different output reactions.

Acknowledgements. The experiments in this paper is carried out by Loïc Bontemps during his final year project in the School of Computer Science, University College Dublin.

References

1. LSTM networks for sentiment analysis. In: LSTM networks for sentiment analysis deeplearning 0.1 documentation. http://deeplearning.net/tutorial/lstm.html#lstm. Accessed 25 Jun 2016
2. DARPA intrusion detection evaluation. (n.d.). http://www.ll.mit.edu/ideval/data/1999data.html. Accessed 30 June 2016
3. Ahmed, M., Mahmood, A.N., Hu, J.: A survey of network anomaly detection techniques. J. Netw. Comput. Appl. **60**, 19–31 (2016)
4. Bhattacharyya, D.K., Kalita, J.K.: Network Anomaly Detection: A Machine Learning Perspective. CRC Press, Boca Raton (2013)
5. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surv. (CSUR) **41**(3), 15 (2009)
6. Chmielewski, A., Wierzchon, S.T.: V-detector algorithm with tree-based structures. In: Proceedings of the International Multiconference on Computer Science and Information Technology, Wisła (Poland), pp. 9–14. Citeseer (2006)
7. Hawkins, S., He, H., Williams, G.J., Baxter, R.A.: Outlier detection using replicator neural networks. In: Kambayashi, Y., Winiwarter, W., Arikawa, M. (eds.) DaWaK 2002. LNCS, vol. 2454, pp. 170–180. Springer, Heidelberg (2002)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
9. KDD Cup Dataset (1999). http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
10. Lee, W., Stolfo, S.J.: A framework for constructing features and models for intrusion detection systems. ACM Trans. Inf. Syst. Secur. (TiSSEC) **3**(4), 227–261 (2000)
11. Lu, W., Ghorbani, A.A.: Network anomaly detection based on wavelet analysis. EURASIP J. Adv. Sig. Proc. **2009**, 4 (2009)
12. Malhotra, P., Vig, L., Shroff, G., Agarwal, P.: Long short term memory networks for anomaly detection in time series. In: Proceedings, p. 89. Presses universitaires de Louvain (2015)
13. Marchi, E., Vesperini, F., Eyben, F., Squartini, S., Schuller, B.: A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional lstm neural networks. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1996–2000. IEEE (2015)
14. Marchi, E., Vesperini, F., Weninger, F., Eyben, F., Squartini, S., Schuller, B.: Non-linear prediction with lstm recurrent neural networks for acoustic novelty detection. In: 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1–7. IEEE (2015)

15. Olsson, T., Holst, A.: A probabilistic approach to aggregating anomalies for unsupervised anomaly detection with industrial applications. In: FLAIRS Conference, pp. 434–439 (2015)
16. Salama, M.A., Eid, H.F., Ramadan, R.A., Darwish, A., Hassanien, A.E.: Hybrid intelligent intrusion detection scheme. In: Gaspar-Cunha, A., Takahashi, R., Schaefer, G., Costa, L. (eds.) Soft Computing in Industrial Applications. AISC, vol. 96, pp. 293–303. Springer, Heidelberg (2011)

A Novel Encryption Mechanism for Door Lock to Resist Jam–and–Relay Attack

Bao An Tran and Viet-Hong Tran(✉)

Faculty of Mechanical Engineering, Department of Mechatronic Engineering,
Ho Chi Minh City University of Technology – VNU-HCM, Ho Chi Minh City, Vietnam
hitmanhostline@gmail.com, tvhong@hcmut.edu.vn

Abstract. This paper describes a novel encryption mechanism for a smarthome lock controlled by smartphone using auto-changing encrypted public key to resist "jam and relay attack". This method is still based on rolling code mechanism and makes use of RSA algorithm to encode the public key so that the key is automatically changed at each time of access. This mechanism will make the security vulnerability of several equivalent keys for each access in rolling code mechanism is no longer be exploited. Furthermore, this method creates a new function to the lock: providing one-time-access-key for visitors. The effective mechanism of resisting jam-and-relay attack is proved and the experiment results show that it is still practical with limited resource system.

Keywords: RSA · Encryption · Smartphone · Smartlock · Jam and relay attack · Replay attack

1 Introduction

Since first mechanical lock was found in ancient Egypt, security is changing day by day to become safer and more convenient. That was the 1950s when the first garage radio remote was invented. At the beginning, it was simply an on/off signal to open. Hence, everyone could open the others. In 1970s, it developed by adding combination into the transmitter. Soon, people recognized that they need privacy. At that time, a large and complex open signal was added. However, these fixed signal codes met a critical disadvantage which people could copy the open signal and resent it later (replay attack). Therefore, this fixed remote code has reached a dead end [1].

Later, in 1990s, "rolling code" or "hopping code" was invented and has been changing overtime. Based on its property that receiver allows several open signals of a transmitter, hacker needs a high performance calculating system and long time to break it. However, at 23rd DEFCON conference, Samy Kamkar announced a cheap mobile device that can hack Microchip's Keeloq algorithm and Texas Instruments's Hisec chip [2]. The question is: "If there is a cheap mobile device that can hack the rolling code, how long will it take to appear a specialized hacking device on the black markets?"

On the other hand, smarthome nowadays is a growing concept. It has become a trend of modern society. This is an idea that everything such as electrical devices or household applicants will be connected together to form a network and the users can use just a

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 153–163, 2016.
DOI: 10.1007/978-3-319-48057-2_10

single personal device such as a smartphone or laptop to control these devices [3]. As a consequence, many companies try to invent new devices which can be controlled by application on smartphones or laptops. Remote-controlled lock, therefore, brings people convenience in smarthome by utilizing remote or phone, but it is necessary to have a better mechanism to replace Keeloq algorithm.

This paper proposes a novel mechanism by applying an asymetric encryption to encode the public key so that there is only key per access. This mechanism will block the security hole of Keelog algorithm, and also provide a new function to the lock: owner can give a visitor an one-time-access key to temporarily open the door.

2 Background

2.1 Asymmetric Encryption

Asymmetric key encryption or public-key encryption, on the other hand, makes use of two keys: a private key and a public key. The public key is used for encrypting, while the private key is used for decrypting. Two of the most widely used asymmetric key algorithms are: RSA and ECC [4].

- Advantages: The encryption key is published for anyone to use and encrypt messages. However, only the receiving party has access to the decryption key that enables messages to be read [5].
- Disadvantages: Compare to symmetric encryption, it requires lots of resources. It needs a strong hardware to generate the keys because it is based on mathematical calculation. The encryption can be broken if it lasts a sufficient amount of time for criminals to decrypt data based on mathematical processing [4].

2.2 Rolling Mechanism

Rolling code (or sometimes called a hopping code, Keeloq) is used in many keyless entry systems to prevent replay attacks where an eavesdropper records the transmission and replays it at a later time to cause the receiver to unlock.

At 23rd DEFCON – largest annual underground hacking conference, Samy Kamkar claim to successfully hack Nissan, Cadillac, Ford, Toyota, Lotus, Volkswagen car's brand; as well as garage of Genie and Liftmaster. From Samy's perspective, the root cause is Keeloq algorithm. By exploiting vulnerability of Keeloq algorithm that the

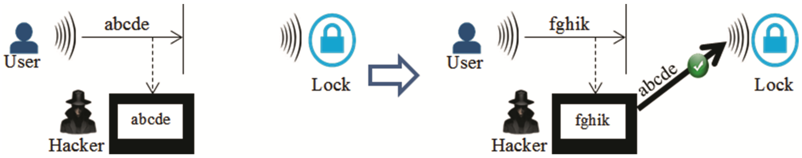


Fig. 1. Jam and relay attack

receiver has to accept more key than the key sent by transmitter, hacker may block one signal sent to the lock (jam), then store that key to unlock victim's lock later (relay). Figure 1 shows the mechanism of jam-and-relay attack.

3 Proposed Security Mechanism

3.1 Auto-changing Code - Key Asymmetric Encryption Mechanism

The block diagram of proposed security mechanism is shown in Fig. 2. Because the "key" to open the door is a remote or smartphone which is not so good for large computations, most of the heavy computations lays on the circuit on the lock, such as decryption, and key generation. The password is encrypted with a stored public key before sending to the lock. The lock will do two jobs at the same time: (1) decrypt the signal to get the original password and determine to open the door with a correct password; (2) generate a new pair of public key and private key which the new public key will be sent to the remote to store there for the next access. Therefore, the public key is changed after each access. This is similar to a token or cookie that is only valid for each access.

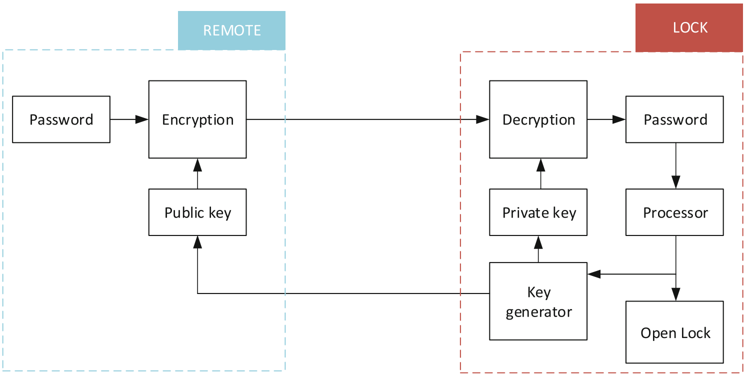


Fig. 2. Auto-changing code mechanism

In practice, we need to choose the encryption algorithm. The mechanism is rather safe by itself, so there is no need to choose a strong encryption algorithm. We propose to use RSA because RSA is often used to pass encrypted shared keys which in turn can perform bulk encryption-decryption operations at high speed. The lock generates n and e as public key then sends it to the remote which will be saved in order to encrypt the password entered from the keypad in the next time. Concurrently, the lock generates and saves the private key d. When it receives the ciphertext from the remote, it uses the private key to decrypt the ciphertext into password then compare it to the standard one to decide whether it is correct or not and open the lock.

Besides that, each lock in a house will need an identification number (ID) to distinguish from each other. The ID of each lock can be a part of public key that only allows determined remote to open. This structure of the public key makes a flexibility in applications. The IDs can be pre–defined and fixed in a closed smarthome system, or generated online in a Internet-of-Things system.

Furthermore, this system can create one-time-access-key for visitors without requiring online system. To be more specified, a visitor at front door can send public key to home owner, who is in office, via text message or the app on smartphone. Then, the owner uses an application to calculate and send visitor ciphertext that can access one time only.

The remaining problem is that the public key must be synced among the "keys" (remotes or smartphones) each time it changed. However, it is very easy to sync via internet nowadays (Wifi, 3G, 4G connections are available almost of the world).

3.2 Remote's Operation

The operational process in the remote can be seen in Fig. 3. Firstly, the remote has n and e from public key sent from lock. Then, it encrypts the password entered from the keypad M (password) into a number C (cipher text) which is equal to $M^e(\mathrm{mod}(n))$. Finally, the number C will be sent back to the lock. The communication method may be bluetooth because the communication distance is close and bluetooth is widely used in personal devices these days.

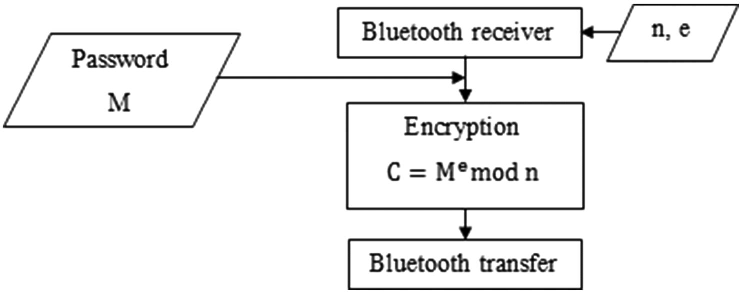


Fig. 3. Remote's process

3.3 Lock's Process

The lock's process consists of key generator and decryption, as illustrated in Fig. 4.

Key Generator.

1. Creating two random prime number p and q.
2. Calculating $n = p \times q$ and $\phi(n) = (p - 1) \times (q - 1)$

3. Generating a number e (encryption exponent) which is coprime with the $\phi(n)$ and $1 < e < \phi(n)$.
4. Finding the number d_P, d_Q and q_{Inv} (decryption exponent) as in Eqs. (1), (2), and (3).

$$e \times d_P \equiv 1(\mathrm{mod}(p-1)) \tag{1}$$

$$e \times d_Q \equiv 1(\mathrm{mod}(q-1)) \tag{2}$$

$$q \times q_{Inv} \equiv 1(\mathrm{mod}\, p) \tag{3}$$

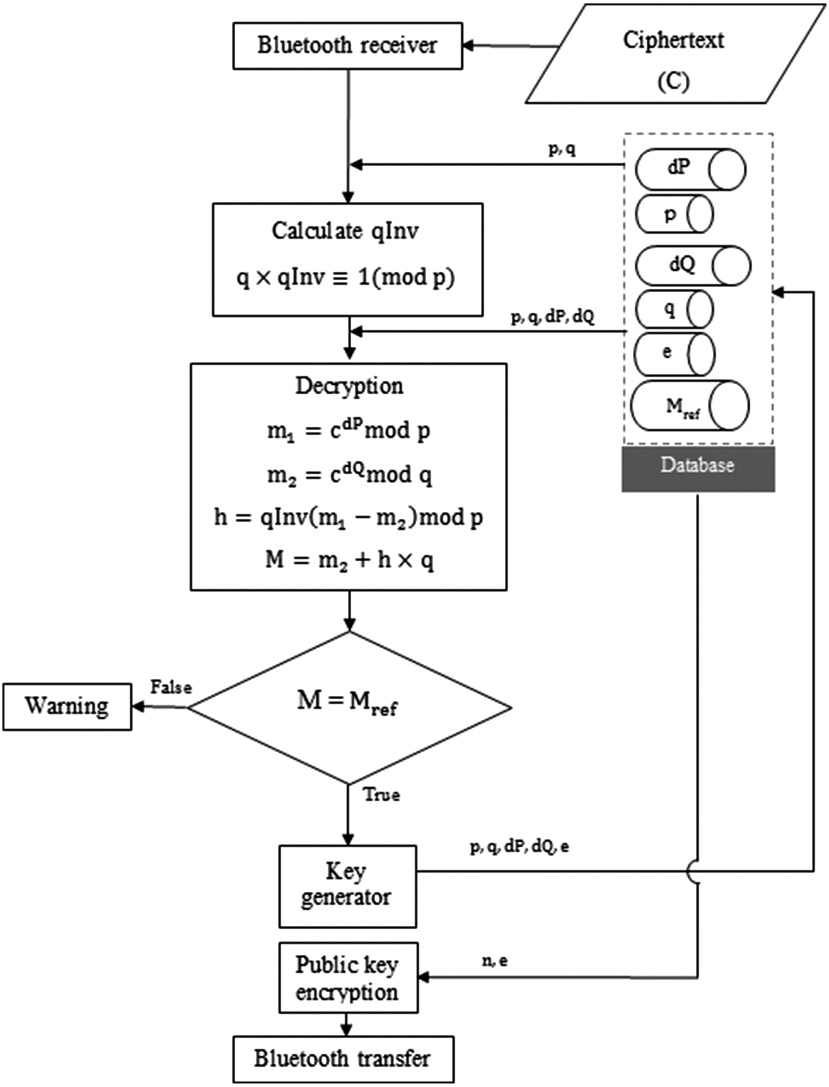


Fig. 4. Lock's process

Decryption. After receiving ciphertext C from remote, the lock starts decryption procedure.

$$m_1 = C^{d_P} \bmod p \tag{4}$$

$$m_2 = C^{d_Q} \bmod q \tag{5}$$

$$h = q_{Inv}(m_1 - m_2) \bmod p \tag{6}$$

$$M = m_2 + h \times q \tag{7}$$

M is then compared to the reference password. If they are equal, door opens.

3.4 Innovative Application of RSA

There are two problems when applying RSA to auto-changing key system are efficient algorithms to (1) generate two big prime numbers p and q; and (2) find coprime e. We propose two algorithms to solve them.

Generate *p* and *q*.

1. An array of prime number is generated. For example, this array contains n elements.

$$A = \left[a_1\, a_2\, a_3 \ldots a_n\right] \tag{8}$$

   Each element in the array A is a prime number.
2. An integer i ($0 < i < n$) will be chosen randomly then the prime number will be created by the following formula.

$$p = \prod_{k=1}^{i} a_k + 1 \tag{9}$$

   p is a prime number because p is not divisible by any integer smaller than $p/2$.
3. Similarly, q is generated by the same method. An integer j ($0 < j < n, j \neq i$) is chosen randomly.

$$q = \prod_{k=1}^{j} a_k + 1 \tag{10}$$

This is the main part for the process of generating public key and private key for RSA encryption. The size of the two prime number p and q depends on the size of the array A and the way that the indexes i and j are chosen. Thus, basically, if the size of A is large enough, p and q can be quite big integers to ensure the security of encryption.

Find Coprime *e*. By using p and q generated from above, it is easy to conclude that $\phi(n)$ is divisible for all integers from a_1 to a_l where $l = \max(i, j)$. Thus, the coprime number of $\phi(n)$ can be find from Eq. 11.

$$e = \prod_{i=(l+1)k}^{k \le n} a_i \tag{11}$$

4 Security Check

4.1 Jam-and-Relay Attack Resistance

The proof of ability to resist jam-and-relay attack of the proposed mechanism is shown in Figs. 5 and 6. In Fig. 5, the lock sends data (n', e') including public key (n, e) to make sure only determined user can access and no confliction with other locks. Then, user decrypts the received data and gets public key. The password is encrypted using public key to make a ciphertext to send to the lock. The ciphertext is then jammed by hacker. This is jam attack.

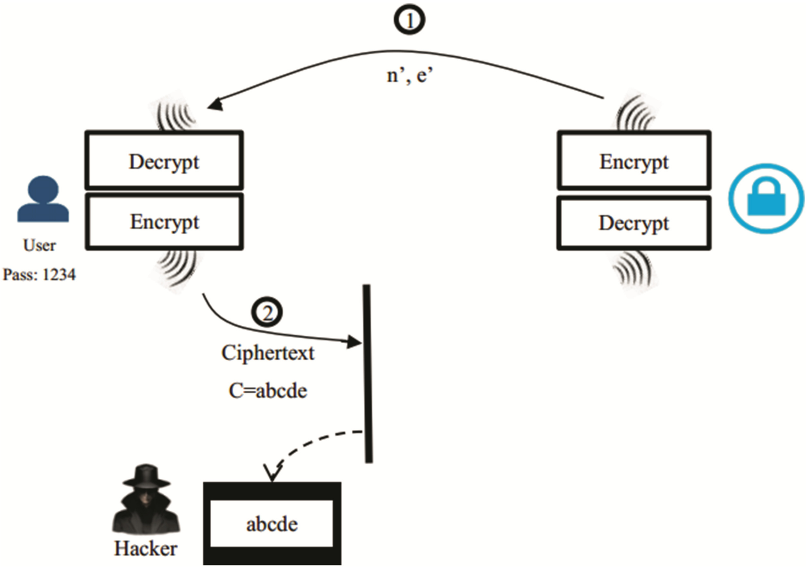


Fig. 5. System under "jam attack"

Because the signal is not reached to the lock, the user has to press the open button again. After successfully open the lock, the public key is changed, so the previous ciphertext that stolen by hacker is useless.

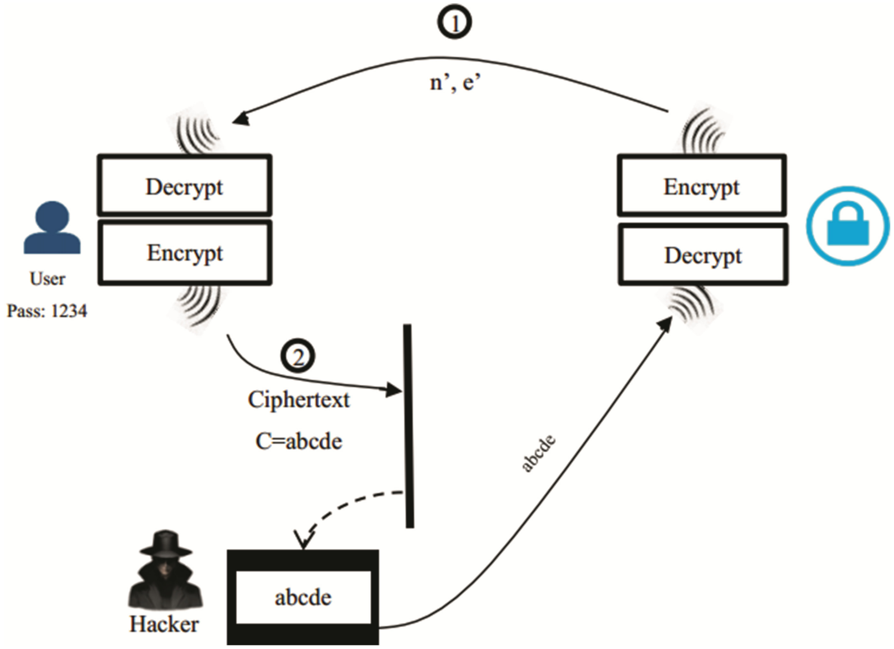


Fig. 6. System under "relay attack"

4.2 RSA Attacking

Another vulnerability of the proposed mechanism is on RSA. Because the decryption is performed inside the lock, it is very hard for attacker to use Timing attack to recover secret key *d*. The feasible way to find *d* is just math–based attacks, but up to present, the methods require large computations. Besides of finding *d*, the method of finding information from ciphertexts shows the best solution, because the ciphertexts change very often. However, it requires attackers a lot of time to collect ciphertexts. Moreover, both public key and private key are changed after each access, so we can conclude that it is impossible to have a simple device to attack the proposed mechanism in near future.

5 Experiment and Results

5.1 Experiment Setup

A prototype is created to verify the mechanism in limited resource system. The lock prototype uses Arduino Mega 2560, which contains ATmega2560 microcontroller, and bluetooth module HC05. The remote is a smartphone Wing VN50, running Android 4.2 (Jelly Bean) with MT6592 CPU (ARM–based, 8 cores, 32-bit, 1.7 GHz). The mobile app is shown in Fig. 7, working on Android environment.

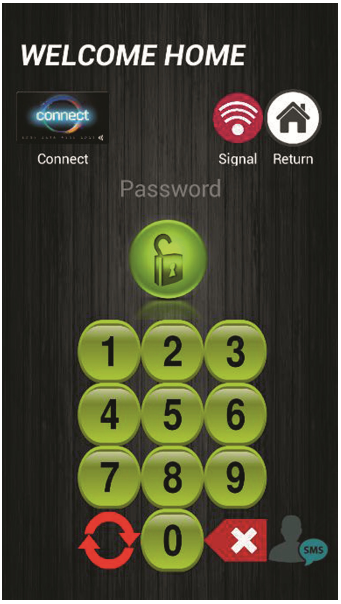


Fig. 7. User interface of door lock application

5.2 Coding

Big Number Problem. The largest number that the microcontroller can handle is 6.8×10^{38} (Double). Therefore, the prototype needs to use BigNumber.h library which had been uploaded by Nick Gammon on Arduino forum in 2012.

Programming. We choose the password length is 4. From RSA properties, the using prime must be smaller than 100. We create a database of prime number containing 12 elements (101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, and 157). Because the database is small, we will choose p and q randomly from this database instead of using our proposed algorithm in Sect. 3.4.

Then e is chosen from 20 prime values (101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197).

From selection above, the prototype can have at least $C_{12}^{2} \times 8 = 528$ values of auto changing ciphertext.

5.3 Results

With the same user password 2210, smartphone sends out different open ciphertexts c (Fig. 8) after each access.

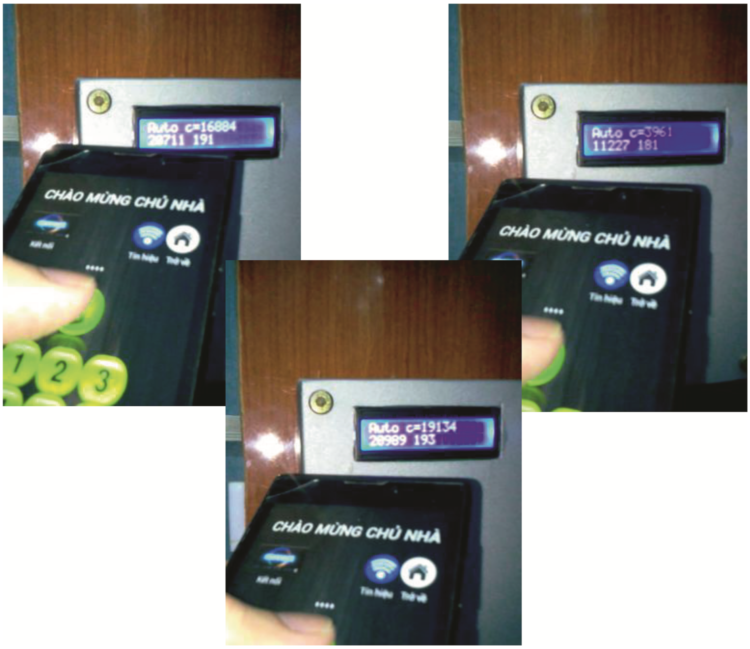


Fig. 8. Three continuously pressing open button with the same password (from left to right: cipher text is 16884, 3961, 19134)

6 Conclusions

This paper proposes an auto-changing code mechanism to against third-party attack ("replay attack" and "jam and relay attack"). The lock sends encrypted public key to the phone. The phone decrypts to acquire public key, then combines these values with password to form ciphertext and sends back to the lock. The lock decrypts this value with secret key to verify the password. If password is true, lock will open and generate new public key and private key.

This paper also proposes to use RSA as the main cryptosystem. RSA always meet problem with big numbers and large computations. However, there are many methods to make it working at high speed such as Chinese Remainder Theorem, and we also propose two methods to generate big prime numbers and calculate public key exponent.

This paper does not analyze deeply to the major factors that cause the complexity in calculations, nor decide the security level of RSA. From prototype, there are two ways to increase security of the RSA design. Firstly, increase the size of prime array. Secondly, increase the size of coprime array. From calculation, increase the size of prime array is more efficient in increasing security of this design.

The possibility of one-time-access key for visitors by giving the ciphertext directly to the visitor is another contribution of the paper.

Acknowledgement. This research is funded by Ho Chi Minh City University of Technology – VNU-HCM, under grant number T-CK-2016-02.

References

1. Brain, M.: How Remote Entry Works. http://auto.howstuffworks.com/remote-entry.htm. Accessed 24 June 2015
2. Kamkar, S.: Drive it like you hacked it: new attacks and tools to wirelessly steal cars. In: Proceedings of 23rd DEF CON Conference (2015)
3. Internet of Things Global Standards Initiative. ITU. Accessed 26 June 2015
4. JSCAPE homepage: a software company focused on offering high-quality, platform-independent and easy-to-use managed file transfer products and services
5. Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: security proofs and improvements. In: Proceedings of International Conference on the Theory and Application of Cryptographic Techniques Bruges, Belgium, pp. 259–274 (2000)

Information and Identity Theft Without ARP Spoofing in LAN Environments

Camilo Albarracin[1], Brayan S. Reyes Daza[2(✉)], and Octavio J. Salcedo Parra[1,2]

[1] Universidad Nacional de Colombia, Bogotá D.C., Colombia
caalbarracinc@unal.edu.co
[2] Internet Inteligente Research Group, Universidad Distrital Francisco José de Caldas,
Bogotá D.C., Colombia
bsreyesd@correo.udistrital.edu.co, osalcedo@udistrital.edu.co

Abstract. Nowadays almost every website use a membership based portal in which each user have a series of permissions that allow him to conduct several actions, in order to check which user has which permissions they use a SessionID this ID allows the web portal to identify the user and grant him the permissions and the information that they need or paid for, this SessionID is sent by HTTP requests so if a third person is able to successfully sniff a package and extract the SessionID, this person will be able to access to the system using the permissions of that member. ARP Spoofing it's a well-known method for sniffing packets although there are effective and easy methods to protect it such as, Static ARP, IDS/IPS systems, port security and other mechanisms of protection. This paper aims to show a technique in sniffing data on a LAN environment without the use of ARP Spoofing in order to be undetectable, unpreventable and effective all focused on Session Hijacking.

Keywords: LAN · ARP · Spoofing · IDS/IPS

1 Introduction

Today most of the people uses Internet both for some simple transactions like entertainment or news and for some slightly more complicated transactions such as purchases, sales, and payments. Thus there are a large number of members and the web portal must recognize and differentiate each of the members, for this it uses something called a SessionID which identifies each user while they are within the website, usually this ID is sent whenever a hidden transaction will be performed and stored on the computer by means of the so-called Cookies, in that way, if at some point a third party can steal the SessionID information, then he could use the identity of the user and obtain full access to the system that they are using.

The use of a LAN structure for Internet is inevitable either in its wired or wireless form, this structure will always be used. The current LAN structures use the ARP Protocol to identify the different entity (hosts) on the LAN, in this way they know who wants to communicate with who and what IP address matches each MAC address, in this way the network card is able to send the data to the correct host, ARP is also

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 164–171, 2016.
DOI: 10.1007/978-3-319-48057-2_11

responsible for identifying the MAC of the router which is the port of departure for all data that will be sent to the Internet, there is an opening of safety in the ARP protocol that allows third parties to perform ARP "Spoof" which allows them to become Man in the middle (MITM) (Sans Institute 2002) and to have access to the data sent on the LAN. LAN structures in general have no mechanism of protection against this attack by default, however, computers can be protected from theft of information using the following methods:

- **Static ARP:** It simply configures each host as well as on the Gateway.
- **The IDS/IPS**: It uses two systems that can detect or prevent potential attacks by monitoring the network.
- **Sniffers Scan:** It finds the network card that is generating a lot of traffic or working in "promiscuous" mode.
- **Security Ports:** Some cisco switch and other high quality brands offer ports security properties for this.

Whenever a user clicks on their Inbox or check an e-mail, the search engine will send HTTP requests containing a specific Cookie which contains the SessionID, with just one of these packages intercepted, it will be able to impersonate the user and access all the privileges that it has.

As we have seen there are many methods to not only block thoroughly these attacks but also to prevent them, all this due to the ARP "Spoof" for this reason it is necessary to find a way to do it without using this technique.

2 Session Spoofing

For the websites with membership (most current) systems, once the user is identified and logged, the portal automatically creates a SessionID and uses it as a reference to know with whom the system is communicating. So when the browser sends a request to the web portal the portal will recognize the user by using the SessionID which is sent through an HTTP request to be confirmed.

Normally the SessionID is encapsulated in a Cookie or in a hidden field, but by being stolen the entire package of information, it does not really matter where it comes encapsulated. Once obtained the SessionID the rest is really intuitive, through editor software packages we will modify our SessionID in the Cookies and we will be able to impersonate the user.

3 ARP "Spoofing"

A third party who is in the same LAN environment as the victim is, or that can remotely access any equipment in the environment has the ability to steal information from the entire LAN using ARP Spoof, ARP Spoof is the process of sending queries/responses ARP to fool the computer of the victim into believing that it is the Router/Gateway, It also sends requests to the Router/Gateway to make them believe that it is the victim's computer by taking advantage of a weak point of the ARP protocol, which is that it does not have an identity verification mechanism (Fig. 1).

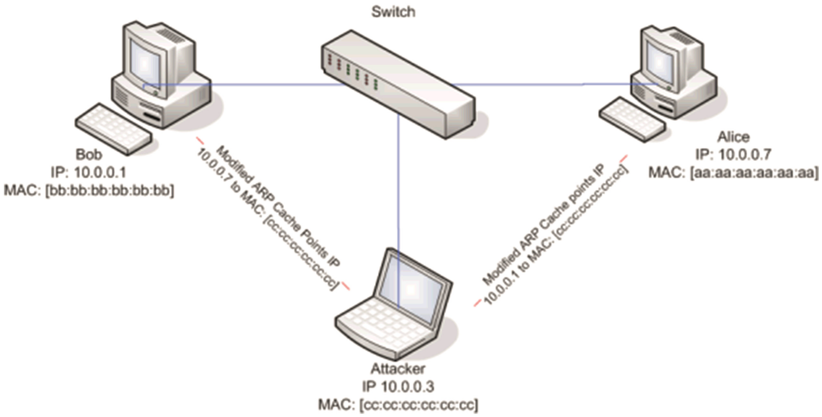

Switch
Bob
IP: 10.0.0.1
MAC: [bb:bb:bb:bb:bb:bb]
Alice
IP: 10.0.0.7
MAC: [aa:aa:aa:aa:aa:aa]
Modified ARP Cache Points IP 10.0.0.7 to MAC: [cc:cc:cc:cc:cc:cc]
Modified ARP Cache points IP 10.0.0.1 to MAC: [cc:cc:cc:cc:cc:cc]
Attacker
IP 10.0.0.3
MAC: [cc:cc:cc:cc:cc:cc]

Fig. 1. ARP Spoofing

Another weak point is the ARP table (ARP Cache) that exist in each host, it refreshes whenever it receives a request or response, thus, although the values of request or response are fraudulent the table will be updated, the data in this table are refreshed every 15 s normally, so the attacker will have to send data flow continuously.

As we have seen the use of static ARP almost completely prevents these attacks since the table is “Fixed” and cannot be modified by an external agent, therefore when the attacker attempts to change the ARP table for this address, this will not allow it.

4 Theft of Information Without Using ARP Spoofing

The problem of ARP Spoofing is that it is easily detectable due to the large influx of data that it needs plus being easily defeatable by the designation of a static ARP table (Veritablelife 2010). However, the packets can be stolen without using this technique, this can be done by fooling the Switch rather than the Gateway and the computer of the victim as in the ARP Spoofing, what we will do is change our MAC and IP address so that they coincide with the Gateway and send packets across the LAN, this will cause that the switch will misunderstand the data and will understand that the Gateway is connected to the port to which we are connected, so when other entities send packets to exit by the Gateway, these packages will actually come to us, there is a problem with this technique and it is that if we send too much traffic on the LAN this will begin to show flaws in its operation so we have to set a time-out in which we cannot intercept information (Fig. 2).

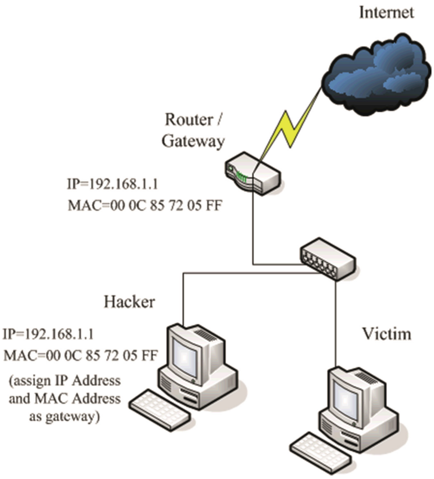
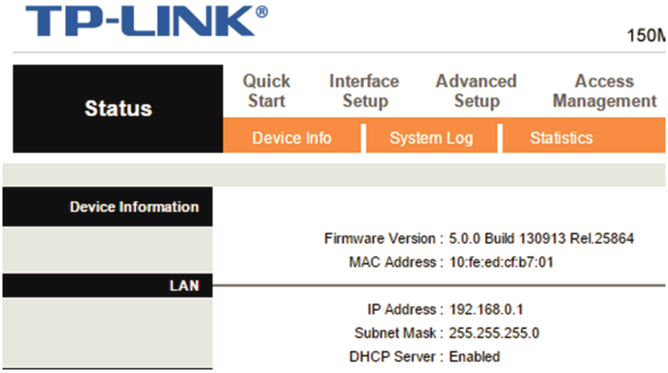


Fig. 2. Without ARP Spoofing

5 Experiment

The taken steps were as follows:

1. Change MAC and IP address of the attacker in such way that they coincide with the router (Fig. 3).



Firmware Version : 5.0.0 Build 130913 Rel.25864
MAC Address : 10:fe:ed:cf:b7:01

IP Address : 192.168.0.1
Subnet Mask : 255.255.255.0
DHCP Server : Enabled

Fig. 3. Setting MAC and IP address

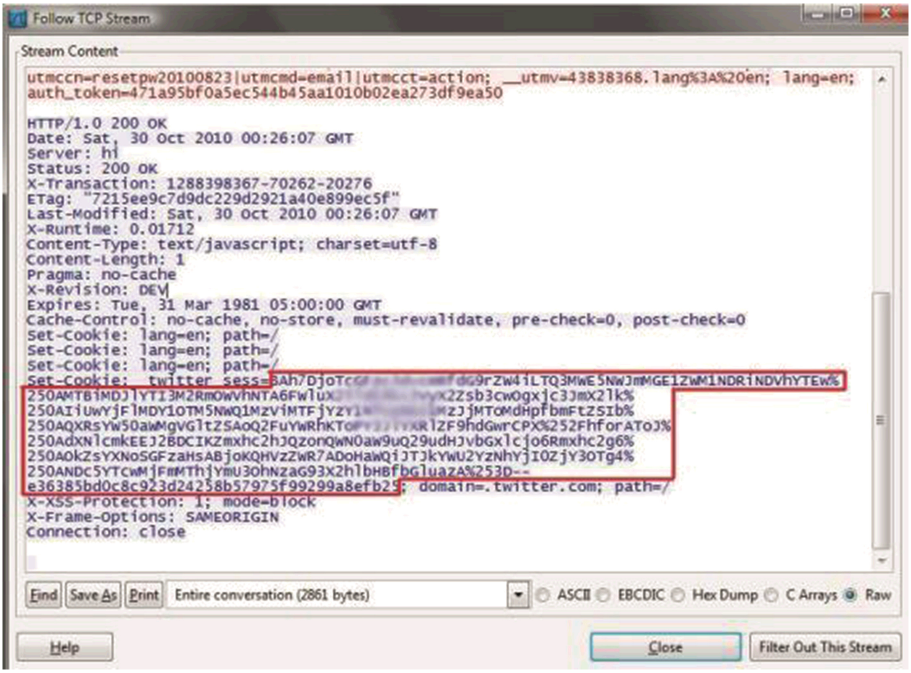
2. To perform a series of "Ping" requests to any entity in the LAN followed by a waiting time to not flood the channel.
3. Steal the information

As we have seen already, if we make requests without waiting time we will generate a DDoS attack and since we want to pass unnoticed, we will execute the request with a prudent intervals of 5 s using the following batch (Fig. 4.).

```
:x
ping 192.168.0.2 -n 1
ping 127.0.0.1 -n 6
goto x
```

Fig. 4. Requesting through ping

Then we shall wait for our victim to make an HTTP request either by clicking in their Inbox, the beginning of their profile on Facebook or simply by try to read his next message, if everything goes as we hope some of the HTTP requests will be forwarded to our machine, and within them we will have a field that keeps the SessionID (Fig. 5), however the victim could detect some sort of slowdown in the system, but with a timeout of five seconds it is so imperceptible that it can be taken as a slow internet or a problem with the computer.



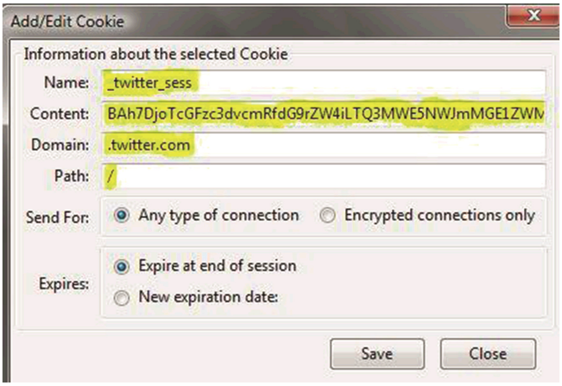
```
utmccn=resetpw20100823|utmcmd=email|utmcct=action; __utmv=43838368.lang%3A%20en; lang=en;
auth_token=471a95bf0a5ec544b45aa1010b02ea273df9ea50

HTTP/1.0 200 OK
Date: Sat, 30 Oct 2010 00:26:07 GMT
Server: hi
Status: 200 OK
X-Transaction: 1288398367-70262-20276
ETag: "7215ee9c7d9dc229d2921a40e899ec5f"
Last-Modified: Sat, 30 Oct 2010 00:26:07 GMT
X-Runtime: 0.01712
Content-Type: text/javascript; charset=utf-8
Content-Length: 1
Pragma: no-cache
X-Revision: DEV
Expires: Tue, 31 Mar 1981 05:00:00 GMT
Cache-Control: no-cache, no-store, must-revalidate, pre-check=0, post-check=0
Set-Cookie: lang=en; path=/
Set-Cookie: lang=en; path=/
Set-Cookie: lang=en; path=/
Set-Cookie: _twitter_sess=BAh7DjoTc[redacted]dG9rZW4iLTQ3MWE5NWJmMGE1ZWM1NDRiNDVhYTEw%
250AMTBiMDJlYTI3M2RmOWVhNTA6FWluX[redacted]vyX2Zsb3cwOgxjc3JmX2lk%
250AIiUwYjFlMDY1OTM5NWQ1MzViMTFjYzY1[redacted]MzJjMTOMdHpfbmFtZSIb%
250AQXRsYW50aWMgVGltZSAoQ2FuYWRhKTo[redacted]XRlZF9hdGwrCPX%252FhforAToJ%
250AdXNlcmkEEJ2BDCIKZmxhc2hJQzonQWN0aW9uQ29udHJvbGxlcjo6Rmxhc2g6%
250AOkZsYXNoSGFzaHsABjoKQHVzZWR7ADoHaWQiJTJkYWU2YzNhYjI0ZjY3OTg4%
250ANDc5YTcwMjFmMThjYmU3OhNzaG93X2hlbHBfbGluazA%253D--
e36385bd0c8c923d24258b57975f99299a8efb25; domain=.twitter.com; path=/
X-XSS-Protection: 1; mode=block
X-Frame-Options: SAMEORIGIN
Connection: close
```

Entire conversation (2861 bytes) ○ ASCII ○ EBCDIC ○ Hex Dump ○ C Arrays ● Raw

Fig. 5. A cookie that contains the SessionID information

As showed in Fig. 5, we have caught between all stolen packages one that contains the SessionID of a site and a specific user, in this case twitter, now the only thing that we have to do is to replicate this cookie in our machine and we will make us go through our victim in this social network, for this we can use a Mozilla Firefox plugin called Cookie Editor and fill in the fields with the necessary information (Fig. 6).



Information about the selected Cookie

Name: _twitter_sess

Content: BAh7DjoTcGFzc3dvcmRfdG9rZW4iLTQ3MWE5NWJmMGE1ZWM

Domain: .twitter.com

Path: /

Send For: ● Any type of connection ○ Encrypted connections only

Expires: ● Expire at end of session ○ New expiration date:

Fig. 6. Creating the fake cookie.

Now if we go to the Twitter.com site will enter under the identity of our victim with all their permissions and no one will be told nor aware. For purposes of measuring the data theft, a follow-up was set to packets that were sent and intercepted by the intruder machine (Table 1).

Table 1. Packages sent vs stolen packages.

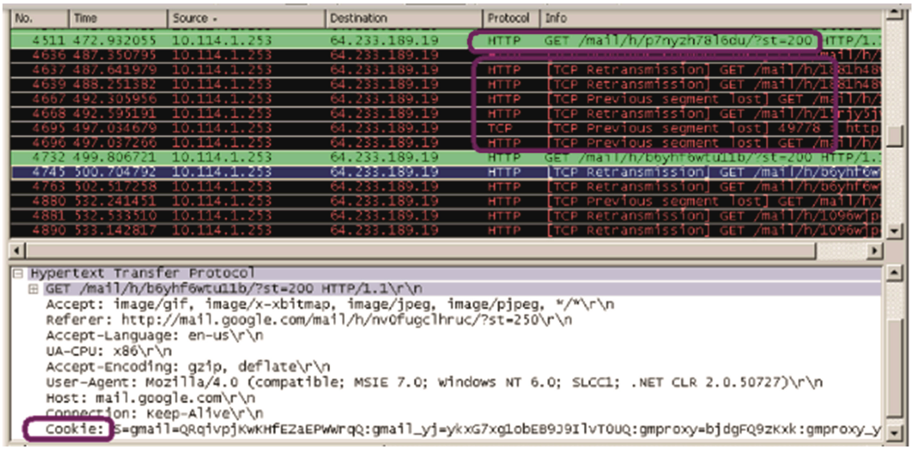
Wait time (s)	Number of packages send	Number of package stolen	Percentage (%)
5	411	219	53.28

It should be noted that with only one stolen package containing the Cookie in which the SessionID is hosted, it is possible to perform the identity theft. Also to measure the impact that the attack causes to the network, we measure the time taken by a connection and the number of failed connection attempts (Table 2).

Table 2. Failed attempts, connection response times.

Wait time (s)	Failed attempts (%)		Response time (ms)	
	Gmail	Hotmail	Gmail	Hotmail
5	0	0	22	18

However, the number of stolen packets is quite high (close to 30 %), taking into account that it is being done with a timeout of five seconds; when analyzing the stolen data TCP retransmission were found (Fig. 7).



```
No.   Time        Source -       Destination     Protocol Info
4511 472.932055  10.114.1.253   64.233.189.19   HTTP     GET /mail/h/p7nyzh78l6du/?st=200 HTTP/1.
4636 487.350795  10.114.1.253   64.233.189.19                                            mail/h/
4637 487.641979  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/181h48
4639 488.251382  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/181h48
4667 492.305956  10.114.1.253   64.233.189.19   HTTP     [TCP Previous segment lost] GET /mail/h/
4668 492.595191  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/1rjy5j
4695 497.034679  10.114.1.253   64.233.189.19   TCP      [TCP Previous segment lost] 49778 > http
4696 497.037266  10.114.1.253   64.233.189.19   HTTP     [TCP Previous segment lost] GET /mail/h/
4732 499.806721  10.114.1.253   64.233.189.19   HTTP     GET /mail/h/b6yhf6wtu11b/?st=200 HTTP/1.
4745 500.704792  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/b6yhf6w
4763 502.517258  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/b6yhf6w
4880 532.241451  10.114.1.253   64.233.189.19   HTTP     [TCP Previous segment lost] GET /mail/h/
4881 532.533510  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/1096wjp
4890 533.142817  10.114.1.253   64.233.189.19   HTTP     [TCP Retransmission] GET /mail/h/1096wjp

⊟ Hypertext Transfer Protocol
  ⊞ GET /mail/h/b6yhf6wtu11b/?st=200 HTTP/1.1\r\n
    Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*\r\n
    Referer: http://mail.google.com/mail/h/nv0fugclhruc/?st=250\r\n
    Accept-Language: en-us\r\n
    UA-CPU: x86\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727)\r\n
    Host: mail.google.com\r\n
    Connection: Keep-Alive\r\n
    Cookie: S=gmail=QRqivpjKwKHfEZaEPWwrqQ:gmail_yj=ykxG7xg1obEB9J9IlvTOUQ:gmproxy=bjdgFQ9zKxk:gmproxy_y
```

Fig. 7. Found TCP retransmissions

Nearly 60 % of the stolen data corresponded to retransmissions of TCP Protocol and even though we can also find the private SessionID Cookies, it is an indication that can alert the victim of their network being under attack.

6 Discussion

To conduct a short analysis and review the obtained data from researches already carried out, we will use the data obtained by (Chomsiri 2008) in a series of similar experiments performed on high safety equipment and different brands with different standards of quality, price and performance. The results of his experiment are shown in the following two Tables 3 and 4.

Table 3. Comparison of stolen packets among the three brands

Wait time (s)	Percentage of data stolen (%)		
	Cisco	3Com	SMC
5	25.9	29	2.4

Table 4. Comparison of response time and failed attempts of connection among the three brands.

Wait time (s)	Response time (ms)			Failed attempts (%)		
	Cisco	3Com	SMC	Cisco	3Com	SMC
5	16	20	21	10	0	0

After a small analysis of the data thrown by (Chomsiri 2008). We see how even though cisco switches are more ‘intelligent’ and expensive they offer nearly the same percentage of theft of data that the 3COM brand and the absolute winner is SMC, but across all scenarios they are better in terms of security than a Home switch since the switch/router package which ETB (Empresa de Telecomunicaciones de Bogota) enterprise offers, allowed us to reach an alarming 53.8 % of stolen packages without any failed attempt, although this is justified by the subject of cost and purpose of use, the numbers are alarming in the four cases, we must also emphasize that although the percentage of data stolen in cisco switches is high considering its cost was the only one who threw failed logon attempts which could alert the victim and therefore could demonstrate that it has slightly better security.

7 Conclusions

The packages can be stolen even without performing an ARP "Spoof", simply by using the change of the IP and MAC address of the machine so that it coincides with the Router/Gateway and send packets to confuse the Switch and then it will send us packages that will be output through the gateway to us, the likelihood of successfully stealing a package is approximately 30 % counting the retransmission packages and this probability depends on the waiting time as on the switch brand.

The victim's machine cannot detect attacks by Sniffers scanning, also the IDS or the IPS will not be able to find this method of stealing information and the use of static ARP tables is also useless because these tables are in the Gateway and the host, the only visible effect of this attack is a delay in the response time on the LAN and the emergence of TCP retransmissions in an irregularly way.

Contrary to what it might be thought, although the cost and the switch technology are quite superior, these features do not have a strong impact on the percentage of stolen data, fact that was demonstrated by experiments conducted by (Chomsiri 2008). And as the last and most important conclusion, the public networks are very dangerous to make important transactions, we never know who else has access to them, who may be attacking them and maybe accessing our information behind our back, it is important that we realize this risk and that we take appropriate safety measures depending on the sites that we want to access.

References

Sans Institute.: SSL Man-in-the-Middle Attack (2002). https://www.sans.org/reading-room/whitepapers/threats/ssl-man-in-the-middle-attacks-480

Chomsiri, T.: Sniffing packets on LAN without ARP Spoofing. In: Third International Conference on Convergence and Hybrid Information Technology, ICCIT 2008. IEEE Xplore (2008)

Veritablelife: Session Hijacking Tutorial (2010). http://www.veritablelife.com/2010/10/29/session-hijacking-tutorial/

Noiumkar, P.: Top 10 free web-mail security test using session Hijacking. In: Proceeding of International Conference on Convergence and hybrid Information Technology, Busan, Korea (2008)

Song, D.: ARP Spoof. https://sourceforge.net/projects/cookie-monster/files/

Data Protection and Data Hiding

A Watermarking Framework for Outsourced and Distributed Relational Databases

Sapana Rani[1(✉)], Dileep Kumar Koshley[1], and Raju Halder[1,2]

[1] Indian Institute of Technology Patna, Patna, India
{sapana.pcs13,dileep.pcs15,halder}@iitp.ac.in
[2] HASLab, INESC TEC, Braga, Portugal
raju.halder@inesctec.pt

Abstract. Unlike centralized databases, watermarking of distributed databases faces serious challenges for various reasons, *e.g.* (*i*) Distribution of data (*ii*) Existence of replication (*iii*) Preservation of watermarks while partitioning and distributing databases, etc. In this paper, we propose a novel watermarking technique for distributed relational databases considering a generic scenario that supports database outsourcing and hybrid partitioning. Our approach addresses the above challenges in an effective way by maintaining meta-data and by making the detection phase partition independent. To the best of our knowledge, this is the first proposal on watermarking of distributed relational databases that supports database outsourcing and its partitioning and distribution in a distributed setting.

Keywords: Watermarking · Distributed databases · Security

1 Introduction

Enormous amount of data is being generated day by day due to the rapid development of internet-based technologies. This huge data need to be stored and managed effectively. Distributed database system is one of the best solutions aiming at improving data-sharing, local autonomy, availability, reliability, performance, etc. [18]. To achieve all these, distributed system divides databases into various partitions (fragments) and stores them physically across various locations along with the associated database-applications. These locations are interconnected by means of communication networks. In recent time, there is a trend to outsource databases, as a cost effective solution, to third party who has required resources to support such distributed settings. This can be understood as three level hierarchy depicted in Fig. 1.

Database contents are always prone to various threats, *e.g.* illegal reselling, ownership claim, tamperation, copyright infringement, etc. [1]. As an effective solution, database watermarking has emerged as a promising technique to detect or prevent such kind of threats. This embeds some kind of information (known as watermark) into data of the database using a secret key which is extracted

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 175–188, 2016.
DOI: 10.1007/978-3-319-48057-2_12

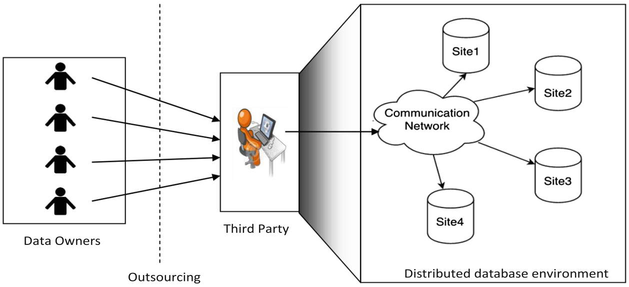


Fig. 1. Three-level hierarchy of distributed database systemThree-level hierarchy of distributed database system

later to reason about a suspicious database. Figure 2 depicts watermark embedding and detection process where a watermark W is embedded into the original database using a private key K (known only to the owner) and later the detection process is performed on any suspicious database using the same private key K by extracting and comparing the embedded watermark (if present) with the original watermark information [4].

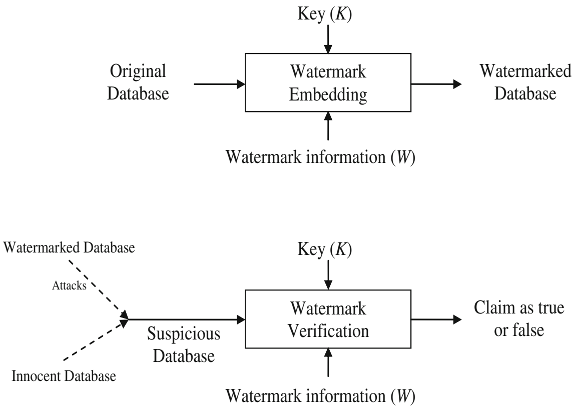


Fig. 2. Basic watermarking technique [4]

Like centralized databases, distributed databases also suffer from all the above-mentioned threats. However, the existing watermarking frameworks for centralized databases [1,9,13,14] are not directly applicable to address those threats in case of distributed databases for the following reasons: (*i*) Distribution of data (*ii*) Existence of replication (*iii*) Preservation of watermarks while performing partitioning and distribution by third party, etc. To the best of our knowledge, till now there is no significant contribution in case of watermarking of distributed relational database systems. Two related works in this direction are found in [6,21]. Authors in [21] proposed a real-time watermarking technique for any kind of digital contents which are distributed among a group of parties in hierarchical manner. However, their proposal has not considered any kind of relational databases and their partitioning over distributed environment. The major drawback is that the data owner has to extract all the watermarks from top to bottom in the hierarchy during verification. Authors in [6], although title refers, have not addressed any challenge in distributed database scenario. In fact, the main technical contributions have not considered any distributed scenario at all.

All the above facts motivate us to propose a novel watermarking technique for distributed database system. The major contributions in this paper are:

- We consider a more generic scenario of distributed relational database systems that supports Database Outsourcing and Hybrid Database Partitioning (*i.e.* both vertical and horizontal partitioning).
- We propose a watermarking framework for distributed databases which allows us to apply any existing suitable centralized database watermarking algorithm separately for each database-partition. However, as the choice of suitable algorithm depends on many factors of each partition (*e.g.* capacity, cover type, etc.), we keep this out of the scope of this work.
- We consider key management scheme in such a way to make the watermarked database more robust *w.r.t.* various threats.
- Most importantly, our proposal aims at making the embedded watermark safe *w.r.t.* database partitioning and distribution by third party. Moreover, the detection phase is completely partition-independent.

This is worthwhile to mention that our approach is suitable for static partitioning and infrequent dynamic partitioning [22], where in the later case a re-watermarking is necessary to make the detection partition-independent.

The structure of the paper is as follows: Sect. 2 describes briefly the existing watermarking techniques in the literature. Section 3 discusses the proposed watermark embedding and detection technique for distributed databases. The experimental results are discussed in Sect. 4. Finally we conclude our work in Sect. 5.

2 Related Works

A series of works on watermarking of centralized databases has been proposed for last 15 years [1,5,7,8,12–15,19,20]. A comprehensive survey can be found in [9]. Among these, a large number of proposals [1,13,19,25] refer to distortion-based watermarking, whereas many others [3,5,8,12,14] refer to distortion-free watermarking of relational databases. All these techniques consider various attribute-types ranging from numerical, categorical, string, etc. as cover to embed watermarks. Among the recently proposed works, an extensive survey on reversible watermarking techniques for relational database is reported in [12]. These techniques ensure original data recovery from watermarked data. A fragile zero-distortion watermarking technique for textual relational database is proposed in [3]. This scheme is based on local characteristics of the relation itself such as frequencies of characters and text length to generate the watermark aiming at preserving data integrity and data quality. Authors in [5] proposed a new approach based on fragile zero watermarking for the authentication of numeric relational data. Here the database relation is partitioned into independent square matrix groups and the watermark is generated using the determinant and minor of the generated square matrix. To protect integrity of database relations, Khan and Hussain [14] proposed a fragile scheme based on zero watermarking technique extracting the local characteristics of the database content, *e.g.* frequency distribution of digits, lengths, ranges of data values, etc. The proposed technique in [13] embeds each bit of a multibit watermark (generated from date-time) in every selected tuple for having maximum robustness even if an attacker is somehow able to successfully corrupt the watermark in some selected part of the data set.

As already mentioned in the previous section, authors in [6,21] proposed watermarking technique in the context of distributed relational databases. However, they have not considered the core properties of distributed scenario during watermark embedding and detection. To be more precise, [21] considers digital contents which are distributed among a group of parties in hierarchical manner. Similarly, the main technical contributions in [6] have not considered any distributed scenario at all.

3 Proposed Watermarking Technique

In this section, we propose a novel watermarking technique for distributed databases. The proposal is based on the scenario where database owner outsources data to a third party as depicted in Fig. 1, assuming that third party has required resources to manage it. Let us describe in detail each of the phases of our proposed watermarking technique.

3.1 Watermark Embedding

The watermark embedding phase consists of the following three phases:

Phase 1: Initial exchange of partition information.

Data owner will initiate this process to exchange some basic information with the third party in order to obtain some initial information about the partitioning and distribution of the database.

Let DB_schema be a relational database schema. Let INF be a set of specifications and requirements about the database and its associated applications, which must be preserved after partitioning and distribution by the third party. For example, INF may include confidentiality and visibility constraints [24], user access information [18], query behaviours [2], etc.

To start this process, the data owner provides DB_schema and INF to the third party. As a result, the third party will send back to the owner a partition overview ψ of the database. This partition overview may be a set of partitions where each partition is either a subset of attributes (vertical partitioning) or a subset of tuples with common properties (horizontal partitioning) or both (hybrid partitioning).

Let us formalize the partition overview ψ. Let t_schema be a schema of a database relation that belongs to DB_schema. The horizontal partitioning of t_schema is formally represented by $\langle t\_schema, f_h \rangle$ where A is the set of all attributes in t_schema and f_h is a partial function defined over A. For instance, f_h can be a mapping of A to a set of properties represented by first order predicate formulas [10] or any other algebraic functions like hash [2]. The horizontal partitioning of tuples in an instance of t_schema is performed by using f_h where each partition contains tuples with similar properties. Similarly, the vertical partitioning can be formalized by $\langle t\_schema, f_v \rangle$ where $\wp(A)$ is the power set of A and $f_v(A) \subseteq \wp(A)$. Observe that the definitions of f_h and f_v depend on INF in order to satisfy the specifications and requirements. Therefore, in general, the hybrid partitioning is formally defined as $\langle t\_schema, f_h, f_v \rangle$. The partition overview ψ of DB_schema satisfying INF is formally defined as

$$\psi \triangleq \{\ \langle t\_schema, f_h, f_v \rangle \mid t\_schema \in DB\_schema\}$$

Phase 2: Watermarking by data owner.

Given a partition overview ψ (provided by the third party) and a secret key K, the data owner embeds watermark into the original database DB. To this aim, the data owner performs the following two:

- *Key Management*: Obtain a set of n different sub-keys $\{K_i \mid i = 1, 2, \ldots, n\}$ from K where n represents the number of fragments obtained from the partition overview ψ (denoted $|\psi|$), and
- *Watermark Embedding*: Embed the watermark W into DB using n sub-keys.

Let us describe each in detail:

Key Management. As our aim is to make the watermark detection partition-independent, the prime challenge here is to select private key K properly and to watermark the database by using K in such a way that partitioning of the database DB by third party must not affect this watermarking.

Watermarking by the data owner considering the future partitioning (by third party) leads to following four possibilities:

- Same Watermark, Same Key: Embedding same watermark into different partitions using same key.
- Different Watermark, Same Key: Embedding different watermarks into different partitions using same key.
- Same Watermark, Different Key: Embedding same watermark into different partitions using different keys.

– Different Watermark, Different Key: Embedding different watermarks into different partitions using different keys.

In our approach, we consider "Same Watermark, Different Key" scenario in which if somehow the watermark is extracted at one site, it will not expose the watermarks embedded into other database-partitions at other sites. Moreover, this serves the purpose of making watermark detection partition-independent as well.

To achieve our objective, we consider k out of n secret sharing schemes [17,23] which states that the secret key K can be recovered from any set of k shares (where k is a threshold) out of n shares of K. Observe that this reduces the challenges in managing and distributing large number of independent keys for all database-partitions in distributed settings. In our approach, we use Mignotte's scheme as this leads to small and compact shares [11]. Algorithm 1 provides detail steps of the Mignotte's scheme to obtain n shares of secret key. Here $n = |\psi|$ that indicates the number of partitions. We have a secret key K which is partitioned into different shares, $\{K_i \mid i = 1, 2, \ldots, n\}$ that is used in watermarking of various partitions.

Algorithm 1. `KEY-COMPUTATION`

```
Input : Partition overview $\psi$, Secret key $K$
Output : Shares $\{K_i \mid i = 1, 2, \ldots, n\}$ of the secret key $K$
1: Let $n = |\psi|$ and $k$ be a threshold, where $|\psi|$ represents the number of partitions.
2: Choose $n$ pairwise co-prime integers $m_1, m_2, ..., m_n | (m_1 \times ... \times m_k) > (m_{n-k+2} \times ... \times m_n)$.
3: Select secret key $K$ such that $\beta < K < \alpha$ where $\alpha = (m_1 \times ... \times m_k)$ and $\beta = (m_{n-k+2} \times ... \times m_n)$.
4: Compute shares of secret key as $K_i = K \bmod m_i$        // $\forall\ i \in 1$ to $n$.
5: Return $\{K_i \mid i = 1, 2, \ldots, n\}$.
```

Watermark Embedding. Data owner watermarks the database DB using shares $\{K_i \mid i = 1, 2, ..., |\psi|\}$, obtained from the secret key K by using Algorithm 1. Suppose DB^i represents i^{th} database-partition in the partition overview ψ. The distributed watermarking is formalized as:

$$\begin{aligned}
&\texttt{DistWM\_Embed}(DB, \psi, W, K)\\
&\qquad = \bigcup_{i \in 1 \ldots |\psi|} \texttt{WM\_Embed}(DB^i, W, K_i)\\
&\qquad = \bigcup_{i \in 1 \ldots |\psi|} DB^i_w\\
&\qquad = DB_w
\end{aligned}$$

Observe that DB^i is watermarked using the share K_i. Owner may use any existing suitable centralized database watermarking algorithm `WM_Embed`[1] to

[1] The selection of suitable watermarking algorithms is an orthogonal research topic.

watermark each partition DB^i, $i \in 1 \ldots |\psi|$. Once watermarked, data owner then outsources the watermarked database DB_w to the third party.

The overall watermarking process performed by the data owner is summarized in Algorithm 2.

Algorithm 2. `Dist_WM_Embed`

```
Input : Database $DB$, Watermark $W$, Specifications $INF$, Secret key $K$.
Output : Watermarked database $DB_w$.
1: Send $\{DB\_schema, INF\}$ to the third party.
2: Receive a partition overview $\psi$ computed from $\{DB\_schema, INF\}$ by the third party.
3: Generate $n$ shares $\{K_i \mid i = 1, 2, ..., n\}$ of the secret key $K$ using algorithm KEY-COMPUTATION($\psi$, $K$), where $n = |\psi|$.
4: Watermark the database: $DB_w = \bigcup_{i \in 1 \ldots |\psi|}$ WM_Embed($DB^i, W, K_i$), where $DB_i \in \psi$.
5: Send the watermarked database $DB_w$ to the third party.
```

Phase 3: Partitioning and distribution by third party.

Once the third party receives the watermarked database DB_w from the data owner (using Algorithm 2), the third party partitions and distributes DB_w as per ψ. In addition, the third party maintains a metadata table which contains information about the data distribution over the servers. The metadata information consists of partition ID P_i, property description of the data in the partition in the form of first-order formula, the server ID S_j where partition P_i is located, etc. Table 1 depicts a hypothetical example of metadata, where $A_1, \ldots, A_5$ represent attributes.

Table 1. Example of metadata

Partition	Partition description		Server
ID	Schema	Properties	ID
P_1	$\{A_1, A_2, A_3\}$	$A_3 \leq \texttt{avg}(A_3)$	S_3
P_2	$\{A_1, A_4, A_5\}$	$A_4 \leq \texttt{avg}(A_4)$	S_1
P_3	$\{A_1, A_2, A_3\}$	$A_3 > \texttt{avg}(A_3)$	S_2
P_4	$\{A_1, A_4, A_5\}$	$A_4 > \texttt{avg}(A_4)$	S_4

3.2 Watermark Detection

If the data owner finds any suspicious database partition or a part of it (denoted DB_s), he/she will initiate the detection process. The main issue that arises in this phase is how to know the actual key which was used at the time of watermark insertion for DB_s. For this purpose, the data owner communicates with the third party and obtains a partition ID P_i based on the matching of the suspicious database data with the property description of P_i in the metadata table. Once the partition ID P_i is obtained, the owner uses the Mignotte's scheme [17] to obtain i^{th} share K_i from K. This way using K_i the data owner extracts a watermark $W^{'}$ by applying `WM_Detect`(DB_s, K_i) and compares it with the original watermark W. Algorithm 3 formalizes the watermark detection phase. Observe that detection algorithm `WM_Detect` corresponds to the watermark embedding algorithm `WM_Embed`.

Algorithm 3. `Dist_WM_Detect`

```
Input : Suspicious database $DB_s$.
Output : Watermark detection as true or false.
Data owner asks third party for partition ID of the suspicious database $DB_s$.
Third party refers metadata and returns $P_i$ based on the matching of data of $DB_s$ with $P_i$'s property description in metadata.
Data owner computes $i^{th}$ share $K_i$ by applying Mignotte's scheme and extracts watermark $W^{'} = \texttt{WM\_Detect}(DB_s, K_i)$.
If $W^{'} \approx W$, claim := true else claim := false.
```

4 Experimental Analysis

We have performed experiment on a real data set Forest Cover Type [16] that contains 581012 tuples. We have added an extra attribute *id* to the dataset that serves as primary key. The experiment is performed on a server configured with Intel Xeon Processor, 3.07 GHz clock speed, 64 GB RAM, and Linux operating system.

Notations	Description
$Count$	no. of tuples used for particular experiment
ν	no. of attributes used for marking and detection in the relation
γ	fraction of tuples used in the experiment
χ	no. of least significant bit available for marking in an attribute
TC	total count that is marked during embedding
α	significance level of the test for detecting a watermarking
τ	threshold parameter for detecting a watermark
match-count	count of the marks matched successfully during detection

Table 2. Results of watermark embedding

Count	$\|\psi\|$	ν	χ	γ	TC	time (msec)
581012	2	10	15	50	11851	14213425
581012	4	5	15	50	23702	27380395
581012	6	3	15	50	35553	44380295
581012	8	2	15	50	47404	57248920

Table 3. Watermark detection results after random value modification attack in case of 2 partitions of the data-set

Partition		Random modification attack		Detection			
No.of fragments	Count	percent updated	χ-bit updated	match count	τ	time (msec)	Detect?
2	337195	0	NA	6791	3395	4610860	✓
		50	8	6341	3395	4104676	✓
		90	8	5977	3395	4247292	✓
		90	10	4741	3395	4395566	✓
	243817	0	NA	5060	2530	2668006	✓
		50	8	4682	2530	2239336	✓
		90	8	4449	2530	2495227	✓
		90	10	3525	2530	2303253	✓

Table 4. Watermark detection results after random value modification attack in case of 4 partitions of the data-set

Partition		Random modification attack		Detection			
No.of fragments	Count	percent updated	χ-bit updated	match count	τ	time (msec)	Detect?
4	337195	0	NA	6791	3395	3086946	✓
		50	8	6317	3395	3148158	✓
		90	8	5988	3395	3248881	✓
		90	10	4759	3395	3052083	✓
	243817	0	NA	5060	2530	1609398	✓
		50	8	4618	2530	1668649	✓
		90	8	4446	2530	1677063	✓
		90	10	3535	2530	1677395	✓
	280962	0	NA	5753	2876	2192104	✓
		50	8	5509	2876	2099414	✓
		90	8	5320	2876	2208180	✓
		90	10	3990	2876	2198221	✓
	300050	0	NA	6098	3049	2477041	✓
		50	8	5824	3049	2630520	✓
		90	8	5629	3049	2501038	✓
		90	10	4307	3049	2473426	✓

For partition-level watermarking algorithm `WM_Embed`, we have used AHK algorithm [1]. The notations used in our experiment are defined below:

Tables 2 depicts the watermark embedding results (watermark embedding time in millisecond) for various number of partitions in partition-overview ψ.

Table 5. Watermark detection results after random value modification attack in case of 6 partitions of the data-set

Partition		Random modification attack		Detection			
No.of fragments	Count	percent updated	χ-bit updated	match count	τ	time (msec)	Detect?
6	334876	0	NA	5646	2832	2977686	✓
		50	8	5646	2832	2857632	✓
		90	8	5646	2832	2829288	✓
		90	10	5646	2832	2768492	✓
	246136	0	NA	5609	3093	1568444	✓
		50	8	5490	3093	1562750	✓
		90	8	5351	3093	1583879	✓
		90	10	4872	3093	1522225	✓
	371245	0	NA	6290	3165	3588523	✓
		50	8	6290	3165	3611780	✓
		90	8	6290	3165	3473582	✓
		90	10	6290	3165	3479286	✓
	209767	0	NA	4912	2760	1126138	✓
		50	8	4488	2760	1171351	✓
		90	8	4217	2760	1124066	✓
		90	10	4198	2760	1102890	✓
	280877	0	NA	2787	2834	1994684	×
		50	8	2781	2834	2089529	×
		90	8	2809	2834	2009919	×
		90	10	2809	2834	2013371	×
	300135	0	NA	3276	3091	2341871	✓
		50	8	3240	3091	2317585	✓
		90	8	3223	3091	2307532	✓
		90	10	3223	3091	2221183	✓

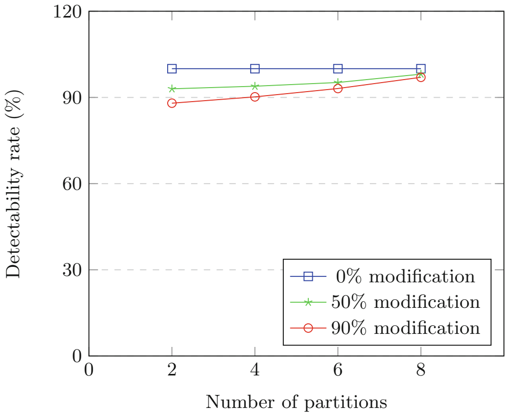


Fig. 3. Watermark detection rate after random value modification attack

Tables 3, 4, 5 and 6 depict watermark detection results (detection time in millisecond, successfully detected or not, etc.) for various number of partitions after random value modification attack took place on 0 %, 50 % and 90 % of the tuples in the partitions.

Table 6. Watermark detection results after random value modification attack in case of 8 partitions of the data-set

Partition		Random modification attack		Detection			
No.of fragments	Count	percent updated	χ-bit updated	match count	τ	time (msec)	Detect?
8	330969	0	NA	6721	3360	2781232	✓
		50	8	6205	3360	2699055	✓
		90	8	5887	3360	2650933	✓
		90	10	4678	3360	2630398	✓
	250043	0	NA	5130	2565	1600756	✓
		50	8	4756	2565	1568929	✓
		90	8	4545	2565	1528359	✓
		90	10	3612	2565	1523670	✓
	337195	0	NA	6791	3395	2790800	✓
		50	8	6791	3395	2795596	✓
		90	8	6791	3395	2768237	✓
		90	10	4761	3395	2775518	✓
	243817	0	NA	5060	2530	1523984	✓
		50	8	5060	2530	1466059	✓
		90	8	5060	2530	1471725	✓
		90	10	3541	2530	1490088	✓
	335646	0	NA	6833	3416	2836748	✓
		50	8	6833	3416	2791317	✓
		90	8	6833	3416	2809398	✓
		90	10	4804	3416	2765616	✓
	245366	0	NA	5018	2509	1484451	✓
		50	8	5018	2509	1465203	✓
		90	8	5018	2509	1508419	✓
		90	10	3489	2509	1480657	✓
	246035	0	NA	4956	2478	1516136	✓
		50	8	4956	2478	1475345	✓
		90	8	4956	2478	1514514	✓
		90	10	3469	2478	1526597	✓
	334977	0	NA	6895	3447	2676237	✓
		50	8	6895	3447	2705557	✓
		90	8	6895	3447	2817721	✓
		90	10	4830	3447	2720335	✓

The rate of watermark detection (=$(match\text{-}count/TC) \times 100$) after performing random value modification attack in each partition is graphically shown in Fig. 3. The observations are summarized below:

- For 0 % value modification, we have 100 % detection rate for all the partitions.
- For 50 % value modification, the rate of detection is 93 %, 93.9 %, 95.2 % and 98.1 % for 2, 4, 6 and 8 partitions respectively.
- For 90 % value modification, the rate of detection is 88 %, 90.2 %, 93.1 % and 97 % for 2, 4, 6 and 8 partitions respectively.
- Therefore, detection rate in presence of random value modification attack increases as we increase the number of partitions.

We have also computed average detection time (in presence of random value modification attack) for all the partitions from Tables 3, 4, 5 and 6 which is

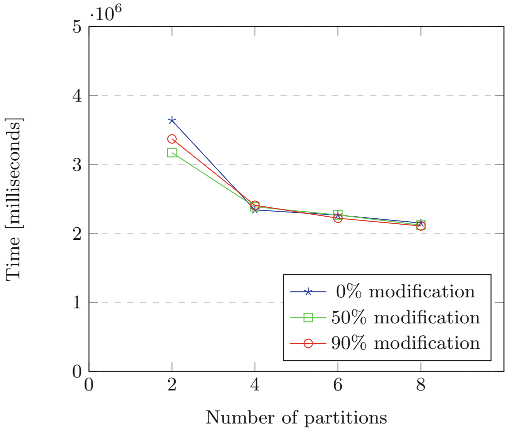


Fig. 4. Average detection time after random value modification attack for $\gamma = 50$

depicted in Fig. 4. Observe that the average detection time decreases as we increase the number of partitions.

5 Conclusions and Future Plans

In this paper, we proposed a novel watermarking technique for distributed database that supports hybrid partitioning. The detection phase in the proposed scheme is partition independent. The key management scheme that we have considered makes the watermark more robust against various attacks, as if anyhow some partitions are attacked, it will not affect the watermarks in other database-partitions. The experimental results show the strength of our approach by analyzing the detection rate with respect to random modification attack. To the best of our knowledge, this is the first work that supports database outsourcing and its partitioning and distribution in a distributed setting. The future works aim to design an efficient watermarking technique for each partition leading to possible improvements in this proposed generic framework and to extend it to the case of big data and cloud computing environment.

Acknowledgment. This work is partially supported by the Council of Scientific and Industrial Research (CSIR), India, and by the ERDF European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme within project “POCI-01-0145-FEDER-006961”, and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia as part of project “UID/EEA/50014/2013”.

References

1. Agrawal, R., Haas, P.J., Kiernan, J.: Watermarking relational data: framework, algorithms and analysis. VLDB J. **12**(2), 157–169 (2003)
2. Agrawal, S., Narasayya, V., Yang, B.: Integrating vertical and horizontal partitioning into automated physical database design. In: Proceedings of the ACM SIGMOD international conference on Management of data, pp. 359–370 (2004)
3. Alfagi, A.S., Manaf, A.A., Hamida, B., Olanrewajub, R.: A zero-distortion fragile watermarking scheme to detect and localize malicious modifications in textual database relations. J. Theor. Appl. Inf. Technol. **84**(3), 404 (2016)
4. Bhattacharya, S., Cortesi, A.: A generic distortion free watermarking technique for relational databases. In: Prakash, A., Sen Gupta, I. (eds.) ICISS 2009. LNCS, vol. 5905, pp. 252–264. Springer, Heidelberg (2009)
5. Camara, L., Li, J., Li, R., Xie, W.: Distortion-free watermarking approach for relational database integrity checking. Math. Probl. Eng. **2014**, 10 (2014)
6. El-Bakry, H.M., Hamada, M.: A developed watermark technique for distributed database security. In: Herrero, Á., Corchado, E., Redondo, C., Alonso, Á. (eds.) Computational Intelligence in Security for Information Systems 2010. AISC, vol. 85, pp. 173–180. Springer, Heidelberg (2010)
7. Farfoura, M.E., Horng, S.J., Wang, X.: A novel blind reversible method for watermarking relational databases. J. Chin. Inst. Eng. **36**(1), 87–97 (2013)
8. Halder, R., Cortesi, A.: A persistent public watermarking of relational databases. In: Jha, S., Mathuria, A. (eds.) ICISS 2010. LNCS, vol. 6503, pp. 216–230. Springer, Heidelberg (2010)
9. Halder, R., Pal, S., Cortesi, A.: Watermarking techniques for relational databases: survey, classification and comparison. J. Univ. Comput. Sci. **16**(21), 3164–3190 (2010)
10. Huth, M., Ryan, M.: Logic in Computer Science: Modelling and Reasoning About Systems. Cambridge University Press, Cambridge (2004)
11. Iftene, S.: General secret sharing based on the chinese remainder theorem with applications in e-voting. Electron. Notes Theor. Comput. Sci. **186**, 67–84 (2007)
12. Iftikhar, S., Kamran, M., Anwar, Z.: A survey on reversible watermarking techniques for relational databases. Secur. Commun. Netw. **8**(15), 2580–2603 (2015)
13. Kamran, M., Suhail, S., Farooq, M.: A robust, distortion minimizing technique for watermarking relational databases using once-for-all usability constraints. IEEE Trans. Knowl. Data Eng. **25**(12), 2694–2707 (2013)
14. Khan, A., Husain, S.A.: A fragile zero watermarking scheme to detect and characterize malicious modifications in database relations. Sci. World J. **2013**, 16 (2013)
15. Khanduja, V., Chakraverty, S., Verma, O.P., Singh, N.: A scheme for robust biometric watermarking in web databases for ownership proof with identification. In: Ślęzak, D., Schaefer, G., Vuong, S.T., Kim, Y.-S. (eds.) AMT 2014. LNCS, vol. 8610, pp. 212–225. Springer, Heidelberg (2014)
16. Forest Cover Type. https://kdd.ics.uci.edu/databases/covertype/covertype.html
17. Mignotte, M.: How to share a secret? In: Beth, T. (ed.) EUROCRYPT 1982. LNCS, vol. 149, pp. 371–375. Springer, Heidelberg (1983)
18. Özsu, M.T., Valduriez, P.: Principles of Distributed Database Systems. Springer, New York (2011)
19. Rani, S., Kachhap, P., Halder, R.: Data-flow analysis-based approach of database watermarking. In: Chaki, R., Cortesi, A., Saeed, K., Chaki, N. (eds.) Advanced Computing and Systems for Security. AISC, vol. 396, pp. 153–171. Springer, Heidelberg (2016). doi:10.1007/978-81-322-2653-6_11

20. Rao, B.V.S., Prasad, M.V.N.K.: Subset selection approach for watermarking relational databases. In: Kannan, R., Andres, F. (eds.) ICDEM 2010. LNCS, vol. 6411, pp. 181–188. Springer, Heidelberg (2012)
21. Razdan, R.: Real-time, distributed, transactional, hybrid watermarking method to provide trace-ability and copyright protection of digital content in peer-to-peer networks (Mar 7 2001), uS Patent App. 09/799,509
22. Rodríguez, L., Li, X.: A dynamic vertical partitioning approach for distributed database system. In: 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 1853–1858. IEEE (2011)
23. Shamir, A.: How to share a secret. Commun. ACM **22**(11), 612–613 (1979)
24. De Capitani di Vimercati, S., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P.: Fragments and loose associations: respecting privacy in data publishing. Proc. VLDB Endowment **3**(1–2), 1370–1381 (2010)
25. Zhou, X., Huang, M., Peng, Z.: An additive-attack-proof watermarking mechanism for databases' copyrights protection using image. In: SAC 2007: Proceedings of the 2007 ACM symposium on Applied computing, Seoul, Korea, pp. 254–258 (2007)

Face Quality Measure for Face Authentication

Quynh Chi Truong[1](✉), Tran Khanh Dang[1], and Trung Ha[2]

[1] Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
{tqchi,khanh}@hcmut.edu.vn
[2] Vietnam National University Ho Chi Minh City,
University of Information Technology, Ho Chi Minh City, Vietnam
trunghlh@uit.edu.vn

Abstract. In a face authentication system, face image quality can significantly influence system performance. Designing an effective image quality measure is necessary to reduce the number of poor quality face images acquired during enrollment and authentication, thereby improving system performance. Furthermore, image quality scores can be used as weights in multimodal system based on weighted score level fusion. In this paper, the authors examined image quality factors, such as contrast, brightness, focus and illumination, and defined quality measure for these factors. The quality measure used template image's, or registration image's, quality as reference quality. Thus, the quality measure does not rely on any reference good quality and criteria to evaluate how good a face image is. The quality measure reflects difference in quality between a template image and a query image. Then, we proposed a face quality measure by combining these factors. Finally, we conducted experiments to evaluate the relationship between face authentication performance and individual image quality factors as well as the combined face quality measure.

Keywords: Face quality index · Face quality measure · Face authentication · Quality metrics

1 Introduction

In biometric systems, the surrounding environment can highly impact on the quality of the input biometric data so that it also affects the system performance [1]. Improving the matching performance does not rely on improving only the matching algorithm but the input biometric data. Poor quality data can decrease the efficiency of the biometric system. Thus, assessing the quality of the input biometric data prior to processing, can be beneficial in terms of improving matching performance.

Moreover, in multimodal systems, qualities of different input modals can play an important role in determining the weight of each modal in the weighted-based fusion scheme. For example, in a multimodal system of two modalities, including face and voice, in order to authenticate to the system, a user must present his face and voice. The system considers that the modality input with a better

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 189–198, 2016.
DOI: 10.1007/978-3-319-48057-2_13

quality can have a better matching accuracy. Therefore, the system will rate the better-quality-input modality higher than the other. The final decision is also highly depended on the better one.

Image quality measures are typically modality specific. There are two categories of quality measures, including generic (used for any biometric modality) and specific (designed to address issues related to a specific biometric modality). However, it is hard to design a on-size-fits-all quality measure for all biometric modalities. Most researches define quality measure for specific modality such as iris [2], fingerprints [3] or faces [4–6]). For the face modality, based on two-dimensional (2D) visible images, generic image quality measures such as average image (AVI) [7], universal quality index (UQI) [8] and IQM [9] can be used. Biometric researchers have also developed modality-specific image quality assessment measures such as those based on redundant wavelets [10].

Several techniques have been proposed in the literature that discuss the benefits of using image quality factors for solving face recognition related problems. However, biometric systems are expected to determine which technique to use to compute a specific quality factor. For example, the sharpness factor can be assessed using several techniques [11,12]. The decision to select one technique over another is problem/application specific and often is made based on experience. However, such a heuristic decision making process becomes even more complicated when multiple image quality factors are considered (sharpness, illumination, focus etc.). Processing time can be saved and face recognition accuracy can benefit from having an alternative solution, that is, a unified technique for computing multiple image quality factors.

The main contributions of this work are: (i) evaluation of various image quality factors for face images, (ii) proposal of a new generic face quality measure.

The rest of the paper is organized as follows: Sect. 2 presents a number of quality factors for face images. Next, we propose a generic face quality measure in Sect. 3. Section 4 is the evaluation of our proposed measure. Conclusions and future works are discussed in Sect. 5.

2 Quality Factors for Face Images

In order to evaluate quality of a face images for authentication, we need to examine various factors, including image's features and posing positions. In scope of this paper, the posing position of face is frontal face. This assumption is quite acceptable in case of mobile authentication in which users have to face to the mobile's cameras. In this paper, we considered four popular image features including brightness, contrast, focus, and illumination.

2.1 Brightness

Brightness is defined as an attribute of a visual sensation according to which a given visual stimulus appears to be more or less intense [13]. The image brightness measure can be calculated as the average of the brightness component after converting it into the HSB color space [14]. HSL stands for hue, saturation, and brightness, and is also often called HSV (V for value)) or HSL (L for lightness).

Another approach is to use YUV color space instead of HSB. Y stands for the luminance component (the brightness) and U and V are the chrominance (color) components. YUV is a color space which encodes a color image taking human perception into account. In our experiments, we calculated image brightness under both HSB and YUV color space. The results showed that image brightness under YUV color space has a higher correlation to the authentication performance. Thus, we calculated image brightness based on YUV color space.

$$B = \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} Y(x, y)}{M.N} \ . \tag{1}$$

where $Y(x, y)$ is the luminance (the brightness) component of a pixel (x, y). The size of image is M × N.

Y is computed from RGB as follows:

$$Y = 0.299R + 0.587G + 0.114B \ . \tag{2}$$

2.2 Contrast

Image contrast is the difference in color intensities of the object and other objects within the same field of view. In face images, contrast shows how a face object distinguishes from the background. A simple way to calculate image contrast is the Michelson contrast equation [15,16]:

$$C = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} \ . \tag{3}$$

where I_{min}, and I_{max} are the minimum and maximum intensity values of the face image.

Another technique to calculate image contrast is the Root Mean Square (RMS) equation [11,17]:

$$C = \sqrt{\frac{\sum_{x=1}^{M} \sum_{y=1}^{N} (I(x, y) - \mu)^2}{M.N}} \ . \tag{4}$$

where μ is the mean intensity value of the face image $I(x, y)$ of size $N \times M$.

Michelson contrast measure does not represent the image contrast factor well due to the fact that it depends only on the maximum and minimum intensity values of the image. Therefore, we used the RMS equation for image contrast. In the experiment, raw images are converted to gray-scale images before calculating contrast values so $I(x, y)$ is in range [0, 256]. The image size is 108 × 108.

2.3 Focus

Image focus refers to the degree of blurring of face images. Image focus can be computed from the energy of the Laplacian which is defined as follows [18]:

$$F = \sum_{x=1}^{M}\sum_{y=1}^{N}(|G_{xx}(x,y)| + |G_{yy}(x,y)|)^2 \; . \tag{5}$$

where G_{xx} and G_{yy} are the second derivatives of the image gradient in the horizontal and vertical directions, respectively.

2.4 Illumination

Luminance distortion is one of the measures of the image factor related to illumination. The term luminance is used to describe the amount of light that passes through or is emitted from a particular area of the image [20]. Image illumination measure is calculated as the weighted sum of the mean intensity values of the image divided into (4–4) blocks [21].

$$I = \sum_{i=1}^{4}\sum_{j=1}^{4} w_{ij}.\bar{I}_{ij} \; . \tag{6}$$

where w_{ij} is the weight factor, and $\bar{I}_{ij}$ is the average intensity value of each block. In a face image, list of weight factors are defined in [21]. The weights are designed so that the middle of image is more considerable than the area near the edge.

3 Proposed Face Quality Measure

A face quality measure should show how good a face image is for an authentication process. In our approach, an image has a good quality if it is taken under a similar condition with the template image. The condition includes brightness factor, contrast factor, focus factor and illumination factor of the image. It means that we compare the conditions in which the two images, e.g. template image and query image, are taken in order to conclude the quality of the query image.

Firstly, we calculated the quality of each factor. Then, we combine them into a single face quality measure.

3.1 Individual Quality Factor Measure

In this section, we used brightness factor for illustration. The other factors work in the same scheme.

Let $B_{template}, B_{query}$ be the brightness values of the template and query images and D_B be the distance of these images.

$$D_B = |B_{template} - B_{query}| \; . \tag{7}$$

The smaller the D_B value is, the better quality is, and vice versa. Then, we normalized the distance in the range [0, 1], where '0' corresponds to bad quality and '1' corresponds to good quality, by using min-max normalization technique. The normalized distance is also the brightness quality measure Q_B.

$$Q_B = 1 - \frac{D_B - D_{Bmin}}{D_{Bmax} - D_{Bmin}} \ . \tag{8}$$

Similarly, contrast quality Q_C, focus quality Q_F and illumination quality measure Q_I are defined.

3.2 Face Quality Measure

We proposed a quality measure for face images which combines individual quality factors in the previous section. Several methods are introduced to combine 'normalized' quality scores [22,23].

- Minimum: $\bar{q} = min(q_1, q_2)$
- Geometric mean: $\bar{q} = \sqrt{q_1.q_2}$
- Difference: $\bar{q} = |q_1 - q_2|$
- Mean: $\bar{q} = \frac{q_1+q_2}{2}$

However, the two quality scores q_1, q_2 may have different effects on the authentication performance. For example, in face authentication, system performance of Principle Component Analysis (PCA) technique [24], which is a common technique to extract feature vectors, is highly affected by brightness. In the evaluation section, the correlation value of brightness is much higher than the correlation values of the other factors. Therefore, to combine quality scores of individual quality factors, we used a weighted average scheme in which correlation values are weights for each quality factors. Correlation value or correlation coefficient reflect relationship between distance of quality and distance of feature vectors from a same person's images. The details of the calculation of correlation coefficients are presented in Sect. 4 (Evaluation).

Let w_B, w_C, w_F, w_I be correlation values of quality factors brightness, contrast, focus, and illumination respectively. We defined a face image quality score Q which is a combination of quality factors as follows:

$$Q = \frac{w_B.Q_B + w_C.Q_C + w_F.Q_F + w_I.Q_I}{w_B + w_C + w_F + w_I} \ . \tag{9}$$

The face image quality measure Q is also in the range [0, 1] where '0' corresponds to bad quality and '1' corresponds to good quality. This proposed quality measure differently treats quality factors of a single biometric feature depending on their correlation values with the authentication performance.

4 Evaluation

In this section, we present various experiments to evaluate: (i) the relationship between individual face quality factor and face authentication performance, and (ii) the effect of the proposed face quality measure to face authentication performance. In our experiments, the face authentication performance is represented by the distance between the template feature vector which is used for registration and the query feature vector which is used for authentication. In general, the expected results of these experiments should confirm that for images from a same person, the better quality score is (e.g. the closer to zero the quality score is), the smaller the distance between template and query feature vectors is (e.g. the higher the authentication performance is).

The algorithm to extract feature vectors from face images is Principle Component Analysis (PCA) [24].

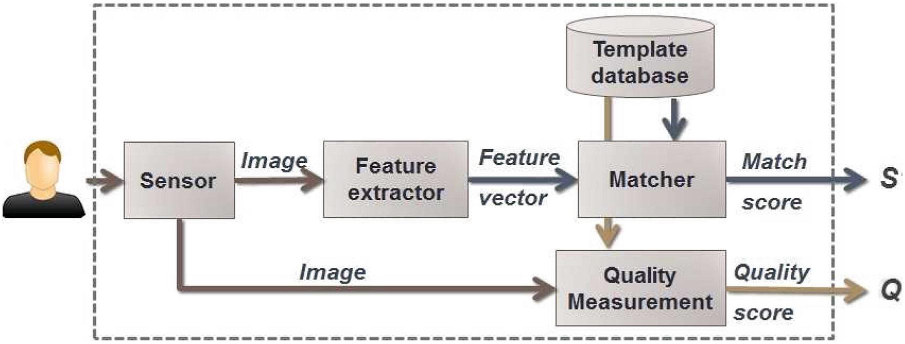


Fig. 1. The evaluation process for face quality

The evaluation process for face quality is carried out as follows:

The first stage: Registration

1- A user registers to the system by supplying his template image. Feature vector is extracted and saved in the template database with user ID.
2- The system calculates quality score of each factor (brightness, contrast, focus, illumination) from the template image and also saves them in the template database.

The second stage: Authentication and Quality Measurement

1- The user presents a query image to the system to authentication. The feature vector of query image is extracted and compared with the feature vector of template image to get match score (the distance between the two feature vectors).
2- The system calculates quality score of each factor (brightness, contrast, focus, illumination) from the query image. Then, the system produces the generic quality measure for the query image by comparing quality scores of the query image to quality scores of the template image.

We repeated the above process for each pair of images of a same person in the face database and then evaluate the correlation coefficient value between the face quality measure and the face authentication performance (Fig. 1).

The face database used for evaluation consists of 1050 frontal face images from 10 people each of whom has 105 images and 44 frontal face images from other people to generate eigen-vectors. In order to evaluate the effects of quality factors to authentication performance, each person is asked to take pictures in a variety of conditions using his/her mobile device.

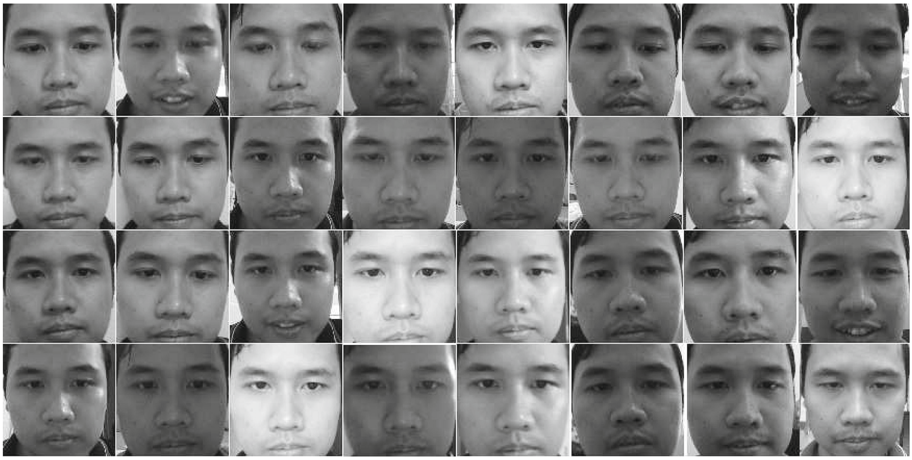


Fig. 2. Some sample images in the testing database

We supposed that a "good" quality measure in authentication should reflect the following assumption: *"For a same person, a query image is classified" good for authentication "if the distance of template and query feature vectors is closed enough to conclude a positive result. Further more, the result should be independent of the algorithm which is used to extract the feature vectors"* (Fig. 2).

In the experiment, for every person, for every pair of his/her images, we calculated a pair of the quality score and the distance of feature vectors. Then we used Pearson correlation coefficient to measure the correlation between the quality scores and the distance of feature vectors. The above process is carried out for all quality factors to get correlation coefficient values. The below table shows correlation coefficient values of brightness, contrast, focus, and illumination factor.

According to these correlation coefficient values, using the PCA algorithm, the brightness factor has the highest impact to the authentication system, while focus and illumination factor have less impact. Please note that, these correlation coefficient values may be changed if the authentication system uses another algorithm to generate feature vectors. These individual quality factor scores become weights in the Eq. 9.

Table 1. Correlation coefficient values of quality measures

Quality measure	Correlation coefficient value
Brightness	0.824
Contrast	0.484
Focus	0.071
Illumination	0.106
Generic face quality measure: mean	0.749
Generic face quality measure: geometric mean	0.605
Generic face quality measure: weighted mean	0.836

Next, we conducted an experiment to evaluate the relationship between the generic face quality measure and face authentication performance in the same way for individual factors. We examined three methods to combine quality factors, including mean, geometric mean and weighted mean (our proposed method). The correlation coefficient values of generic face quality measure are also shown in Table 1. Our proposed quality measure has a highest correlation coefficient values (**0.836**). This result confirms a high correlation between the proposed face quality measure and face authentication performance. Further more, we can conclude that our face quality measure is practical and can be applied in reality.

5 Conclusions and Future Works

This paper presented quality factors for face images and proposed a face quality measure for authentication. The authors' approach is different from previous researches. The proposed quality measure does not rely on any predefined good quality criteria or any reference good images. The face authentication systems, especially mobile face authentication systems, can be deployed in various environments and used different feature extraction algorithms. Therefore, it is hard to define good quality criteria and select a set of reference good images that fit all cases. We used a template image, which a user register to the authentication system, as a reference image to evaluate the quality of the query images from that user. Thus, every user has his/her own reference image. In other words, a query image has a good quality if it is taken under a similar condition with the template image.

In general, our quality measure can be applied for other biometric trails, such as voice, fingerprint, iris, and so on. Thus, the next work is to conduct more experiments on the other biometric trails to confirm the *practical* and *correctness* properties of the proposed quality measure.

After that, we will design a quality-based fusion solution for multi-modal system. The fusion solution will consider quality score of each biometric trail as weighted factor.

References

1. Jain, A., Ross, A., Nandakumar, K.: Introduction to Biometrics. Springer, New York (2011)
2. Zuo, J., Schmid, N.: Adaptive quality-based performance prediction and boosting for iris authentication: methodology and its illustration. IEEE Trans. Inf. Forensics Sec. **2013**(8), 1051–1060 (2013)
3. Merkle, J., Schwaiger, M., Breitenstein, M.: Towards improving the NIST fingerprint image quality (NFIQ) algorithm. In: International Conference on Biometrics Special Interest Group (BIOSIG), Darmstadt, Germany (2010)
4. Hsu, R.L.V., Shah, J., Martin, B.: Quality assessment of facial images. In: Biometric Consortium Conference (BCC), Baltimore, MD, USA (2006)
5. Bhattacharjee, D., Prakash, S., Gupta, P.: No-reference image quality assessment for facial images. In: Huang, D.-S., Gan, Y., Gupta, P., Gromiha, M.M. (eds.) ICIC 2011. LNCS (LNAI), vol. 6839, pp. 594–601. Springer, Heidelberg (2012). doi:10.1007/978-3-642-25944-9_77
6. Wong, Y., Chen, S., Mau, S., Sanderson, C., Lovell, B.: Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Colorado Springs, CO, USA, 2011, pp. 74–81 Won+11, KrD06, WaB02, AdD06, VSN08, Gao+07, Yao+08 (2011)
7. Kryszczuk, K., Drygajlo, A.: On combining evidence for reliability estimation in face verification. In: European Signal Processing Conference (EUSIPCO), Florence, Italy (2006)
8. Wang, Z., Bovik, A.C.: A universal image quality index. IEEE Sig. Process. Lett. **2002**(9), 81–84 (2002)
9. Adler, A., Dembinsky, T.: Human vs. automatic measurement of biometric sample quality. In: IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Ottawa, Canada (2006)
10. Vatsa, M., Singh, R., Noore, A.: SVM-based adaptive biometric image enhancement using quality assessment. In: Prasad, B., Prasanna, S. (eds.) Speech, Audio, Image and Biomedical Signal Processing using Neural Networks. SCI, vol. 83, pp. 351–367. Springer, Heidelberg (2008)
11. Gao, X., Li, S.Z., Liu, R., Zhang, P.: Standardization of face image sample quality. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 242–251. Springer, Heidelberg (2007)
12. Yao, Y., Abidi, B.R., Kalka, N.D., Schmid, N.A., Abidi, M.A.: Improving long range and high magnification face recognition: database acquisition, evaluation, and enhancement. Comput. Vis. Image Underst. **2008**(111), 111–125 (2008)
13. Wyszecki, G., Stiles, W.S.: Color science, Concepts and Methods, Quantitative Data and Formulae. Wiley, New York (2000)
14. Bezryadin, S., Bourov, P., Ilinih, D.: Brightness calculation in digital image processing. In: International Symposium on Technologies for Digital Fulfillment, Las Vegas, NV, USA (2007)
15. Michelson, A.: Studies in Optics. University of Chicago Press, Chicago (1927)
16. Bex, P.J., Makous, W.: Spatial frequency, phase, and the contrast of natural images. J. Opt. Soc. Am. A **19**(6), 1096–1106 (2002)
17. Peli, E.: Contrast in complex images. J. Opt. Soc. Am. A **7**(10), 2032–2040 (1990)
18. Yap, P.-T., Raveendran, P.: Image focus measure based on Chebyshev moments. IEEE Proc. Vis. Image Sig. Process. **151**(2), 128–136 (2004)

19. Pertuz, S., Puig, D., Garcia, M.A.: Analysis of focus measure operators for shape-from-focus. Pattern Recogn. **46**(5), 1415–1432 (2013)
20. Abaza, A., Harrison, M.A., Bourlai, T., Ross, A.: Design and evaluation of photometric image quality measures for effective face recognition. IET Biometrics **3**(4), 314–324 (2014)
21. Abdel-Mottaleb, M., Mahoor, M.: Application notes algorithms for assessing the quality of facial images. IEEE Comput. Intell. Mag. **2**, 10–17 (2007)
22. Grother, P., Tabassi, E.: Performance of biometric quality measures. IEEE Trans. Pattern Anal. Mach. Intell. **29**, 531–543 (2007)
23. Kryszczuk, K., Richiardi, J., Drygajlo, A.: Impact of combining quality measures on biometric sample matching. In: IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS), Washington, DC, USA (2009)
24. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cogn. Neurosci. **3**, 71–86 (1991)

Computer Virus Detection Method Using Feature Extraction of Specific Malicious Opcode Sets Combine with aiNet and Danger Theory

Vu Thanh Nguyen[1], Cao Ngoc Tuan[1](✉), Ly Tan Dung[1],
Vo Minh Hai[2], and Toan Tan Nguyen[1]

[1] University of Information Technology, Vietnam National University,
HCM City, Vietnam
{nguyenvt, toannt}@uit.edu.vn,
{12520478, 12520821}@gm.uit.edu.vn
[2] The Immigration Office of Police Station in HCMC, Ho Chi Minh, Vietnam
haiminh2607@gmail.com

Abstract. Nowadays, many methods of detecting computer viruses are researched towards machine learning and data mining. Among these are the topics related to the automated search algorithm characteristic of the virus. The feature extraction of virus opcode method is proposed in this paper is statistical combinations of x86 machine instruction. The selected instructions are common in a set of virus files and less common in benign files, using some machine learning and data mining algorithms to support. The frequent combination of instruction sets are seen as the operational characteristics of the virus files. Artificial Immune System in combination with Danger Theory will be used for the training of the selected instruction sets into building up a classification system detecting a new file is a virus or not.

Keywords: Feature extraction · x86 opcode · Data mining · Artificial immune network (aiNet) · Danger theory

1 Introduction

Today, in the era of information technology development, virus programs, various types of malicious code have taken advantage of the development to carry out illegal action such as monitoring, data theft, hijacking the host system, … and caused enormous damage to people. There have been many anti-virus programs designed to detect and remove viruses with different algorithms. Several methods have been applied in the identification of the virus characteristics such as code sequences, behavioral or machine learning approaches. They have achieved certain results.

One of the methods for selecting characteristics of virus is to use the data mining algorithms to find out the machine code with the highest frequency [1]. This method will automatically detect and provide frequency machine code. Using the observed level of use of the combination machine code to find out what candidates featured in virus files set.

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 199–208, 2016.
DOI: 10.1007/978-3-319-48057-2_14

The machine learning algorithms inspired by biological research is invested with positive results in many different fields recently. Artificial immune system, one of the theories inspired by natural biological immune system [2] is applied on the issue of detecting computer viruses. Algorithm artificial immune network will be applied to the training steps to create a file classifier which will determine whether a file is a virus or not.

Danger Theory is one of the improvements made in the application of artificial immune system. This theory showed that when using artificial immune system distinguish self cells - immature self we should consider carefully which cell is dangerous or not. This theory will be implemented in order to calculate the risk of abnormal cells if they have the feature of virus or not.

This paper will introduce a new method with artificial immune network algorithm and danger theory as the core. Using the frequency machine code provided by data mining algorithms as training data, artificial immune network algorithm will create a set of detectors with each detector's risk is calculated by danger theory. This set of detector will be used to determine if a file is a virus or not.

2 Related Works

In a paper, Rad and Masrom used statistical analysis techniques to compare the similarity between the virus infected file [3]. They used statistical chart of machine code groups with high prevalence in order to detect the virus variants. And this helps to determine whether a file is a variant of a known virus. There is a similarity with the proposal method in this paper, which also used instruction opcode extraction to classify virus and benign file.

Chao and Tan used the theory of artificial immune system to build up a virus detection system [4]. Another paper by authors Ali and Hussain also has the same idea [9]. They apply methods such as Negative Selection algorithm and Clonal Selection to build the detector. Results showed a positive performance in the detection of viruses. An algorithm of artificial immune system called aiNet is used in this paper as a combination part to generate virus detectors.

Lu et al. proposed the usage of Danger Theory on problems detected viruses on mobile devices [5]. Based on analysis of the danger of local acts, generation and distribution of antibodies in real time, this model can be responsive in preventing and detecting viruses. This paper used Danger Theory to calculate the risk factor of a virus detector, it makes only highest risk value detectors kept remaining in virus or benign final classifier.

3 The Proposed Approach

In this paper, a construction method of detection based on the combination of the stages is proposed as follows: extract combinations of machine code has the highest prevalence of the virus pattern file, then use Artificial Immune System for training to generate the final virus detectors and Danger Theory for calculating risk assessment values for dangerous levels of detectors to choose the optimal candidates.

3.1 Statistic Frequent Opcode Sets

3.1.1 Extraction of x86 Assembly Code

Convert Binary Strings to Machine Instructions. In the first step we need a method to extract the machine code in an executable file (.exe or.com). The method used is to extract the sequence of binary data files in a jump step k and length l. Then proceed to transfer that binary sequence to the corresponding machine code.

Parameters jump step and length selected here are k = 8 and l = 32. This allows the low failure rate in loss of instructions, and a 32-bit length matching the length of a machine code in the MIPS executable file on a windows 32-bit operating system (Fig. 1).

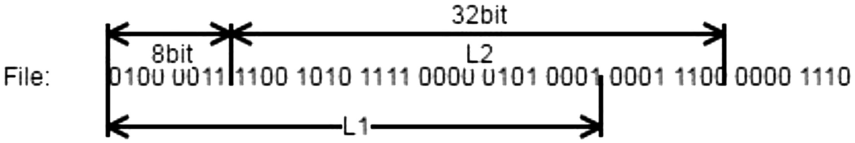
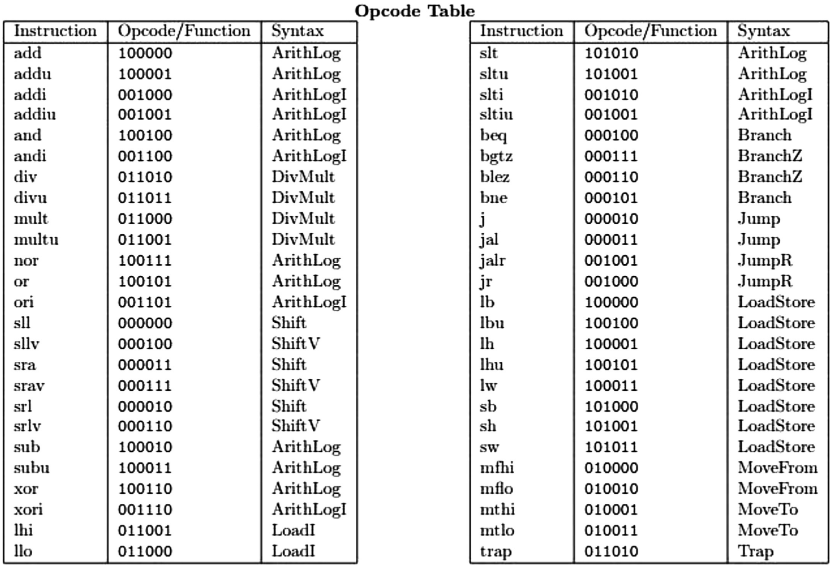


Fig. 1. Extract bit string in an executable file

After a list of 32-bit binary string candidates, the next will conduct its data matching with each instruction in the table of MIPS opcode table. The result will be a corresponding machine code of binary string. If a binary string does not match any MIPS code, it will be removed [6] (Figs. 2 and 3).

Statistic Frequent Opcode Sets in Benign Files Set. After having the MIPS code sets, statistical processes are conducted to see which machine code is common in collected benign files set. If a machine instruction appears more than a minimum support value m_{fb} of benign files do there, then it will be put into frequent specific machine code sets of benign files.

```
Notation
  $I_{benign}$ : benign machine code sets
  $N_{benign}$: number of benign files
  $IF_{benign}$: Frequent benign machine code sets
Begin
  $IF_{benign}$:= $\emptyset$
  For each benign code b do
    If Count(b,$N_{benign}$) >= $m_{fb}$ then
    $IF_{benign}$ := $IF_{benign}$ + b
    End If
  End For
End
```



Opcode Table

Instruction	Opcode/Function	Syntax	Instruction	Opcode/Function	Syntax
add	100000	ArithLog	slt	101010	ArithLog
addu	100001	ArithLog	sltu	101001	ArithLog
addi	001000	ArithLogI	slti	001010	ArithLogI
addiu	001001	ArithLogI	sltiu	001001	ArithLogI
and	100100	ArithLog	beq	000100	Branch
andi	001100	ArithLogI	bgtz	000111	BranchZ
div	011010	DivMult	blez	000110	BranchZ
divu	011011	DivMult	bne	000101	Branch
mult	011000	DivMult	j	000010	Jump
multu	011001	DivMult	jal	000011	Jump
nor	100111	ArithLog	jalr	001001	JumpR
or	100101	ArithLog	jr	001000	JumpR
ori	001101	ArithLogI	lb	100000	LoadStore
sll	000000	Shift	lbu	100100	LoadStore
sllv	000100	ShiftV	lh	100001	LoadStore
sra	000011	Shift	lhu	100101	LoadStore
srav	000111	ShiftV	lw	100011	LoadStore
srl	000010	Shift	sb	101000	LoadStore
srlv	000110	ShiftV	sh	101001	LoadStore
sub	100010	ArithLog	sw	101011	LoadStore
subu	100011	ArithLog	mfhi	010000	MoveFrom
xor	100110	ArithLog	mflo	010010	MoveFrom
xori	001110	ArithLogI	mthi	010001	MoveTo
lhi	011001	LoadI	mtlo	010011	MoveTo
llo	011000	LoadI	trap	011010	Trap

Fig. 2. Opcode MIPS x86 table

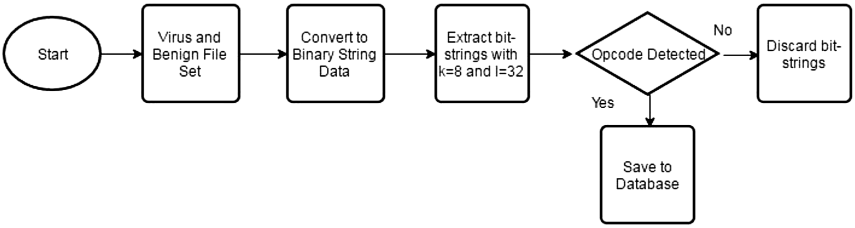

Start
Virus and Benign File Set
Convert to Binary String Data
Extract bit-strings with k=8 and l=32
Opcode Detected
No
Discard bit-strings
Yes
Save to Database

Fig. 3. Convert binary strings to machine instructions flowchart

Remove Frequent Specific Machine Code Sets of Benign Files in Virus Files. Next is going to remove frequent specific machine code sets of benign files in virus files. A computer virus requires a host program to infect. Therefore, an infected file always has two parts: benign code (usually big) and virus code (smaller than host program). This helps the virus files no longer have the benign machine instructions. Keep only those candidates may be specific of the virus files. It is a required step to have a set of full virus opcode in order to get experimental results.

```
Notation
  $I_{virus}$ : virus machine code sets
  $IF_{benign}$: Frequent benign machine code sets
Begin
  For each frequent benign code b do
    If b $\in I_{virus}$ then
      $I_{virus}$ := $I_{virus}$ - b
    End If
  End For
End
```

3.1.2 Statistic Combination of Machine Code Sets

Generate Combination Machine Code Sets Candidates. According to MIPS code from each virus file, the code sequence k-element combinations can be generated with the original order of appearance unchanged. The code sequence will be considered as candidates for the code characteristic of viruses collected files (Fig. 4).

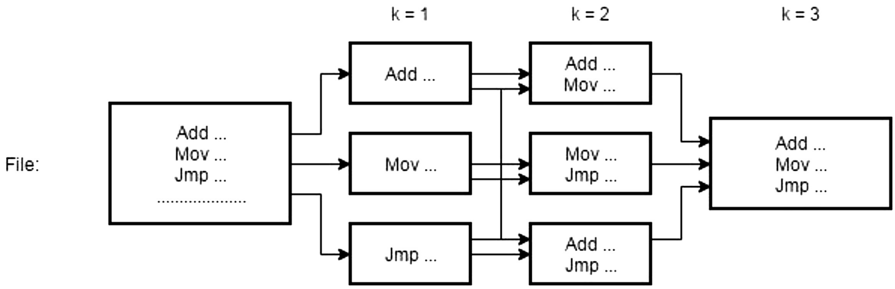


Fig. 4. Generate combination machine code sets candidates of a virus file

If Uf is combination machine code sets of a file f, Ck is k- combination machine codes, then

$$\mathrm{Uf} = \mathrm{C1} \cup \mathrm{C2} \cup \ldots \cup \mathrm{Cn} - 1 \cup \mathrm{Cn}$$

Find out Frequent Combination Machine Code Sets. Using Apriori algorithm [7] in data mining with set of objects is the combination of machine code candidates generated in the above step. After this step, the longest combination of code having frequency greater than minimum support value m_{fc} threshold will be retained as the training specific to the following machine learning steps.

```
Notation
  $U_{virus}$ : virus machine code combination sets
  $N_{virus}$ : number of virus files
  $IF_{cm}$ : frequent machine code combination sets
Begin
  $IF_{cm}$ := Ø
  For each machine code combination c do
    If Apriori(c,$N_{virus}$) >= $m_{fc}$ and FrequentSetParentList(c) == Ø then
      $IF_{cm}$ := $IF_{cm}$ + c
      End If
    End For
End
```

3.2 Artificial Immune System

3.2.1 aiNet Algorithm

In the above steps, there was a complex set of machine code characteristic of viruses collected files. In this step, the training process performed by algorithms aiNet immune system theory was applied due to positive effect in a number of previous case studies.

aiNet algorithm consists of 3 main steps: randomly generated gene candidate (combination of code), perform mutation and replication, ultimately selected the best genes.

Generate Candidate Genes Randomly. With the set of code characteristic of virus files provided above, we follow with conducting randomly generated and combining candidate genes. Implementation method is to choose any length Lg <= maxLg (greatest length of the genes), then randomly select Lg element from the set of machine code of the virus files.

Gene Mutation. To perform candidate gene mutations, the method alternating some random sequences is applied. Alternative sequences will be chosen at random, as long as it is a MIPS code.

Select Best Genes. The best genes will be selected based on affinity (3.2.2) of it with the antigen trained in artificial immune network. This give virus detectors the capability of detecting with the highest accuracy.

aiNet Algorithm. Here is the pseudo code of the algorithms of artificial immune network based on the article [2], with randomly generated steps, copying, mutation and selection are implemented according to the method described above. The concept of antibody, antigen, and B-cell is inspired by Immune System. In the paper, the antibody is the machine code sets which detect virus, the antigen is the machine code sets from virus files which used to trainning detectors.

```
Notation
  S : a set of antigens, representing data elements to be clustered
  nt : network affinity threshold
  ct : clonal pool threshold
  h : number of highest affinity clones
  a : number of new antibodies to introduce.
  N : set of memory detectors capable of classifying unseen patterns

Begin
  Generate set of random specificity B-cells N
  Repeat
   For all antigens ag ∈ S do
     Calculate affinity of all B-cells b ∈ N with ag
     Select highest affinity B-cells, perform affinity proportional cloning, place clones in C
     For all B- cell clones c ∈ C do
       Mutate c at rate inversely proportional to affinity
        Determine affinity of c with ag
      End For
      Select h highest affinity clones c ∈ C and place in D
      Remove all elements of D whose affinity with ag is less than ct
      Remove elements of D whose affinity with other elements in D is less than ct
      Insert remaining elements of D into N.
     End For
     Determine affinity between each pair of B-cells in N
     Systemically remove all B cells whose affinity to another B cell is less than nt
     Introduce a new, randomly generated, B-cells into N
Until a stopping condition has been satisfied
End
```

3.2.2 Affinity

In the theory of artificial immune systems, affinity is a concept that refers to the degree of genetic similarity. There are several methods to calculate the affinity such as the Euclidean distance, Hamming distance… depending on the type of data represented. With current data particular, the combination of genes is machine codes set, this paper proposes a way to measure the similarity of set A and B as follows (Fig. 5).

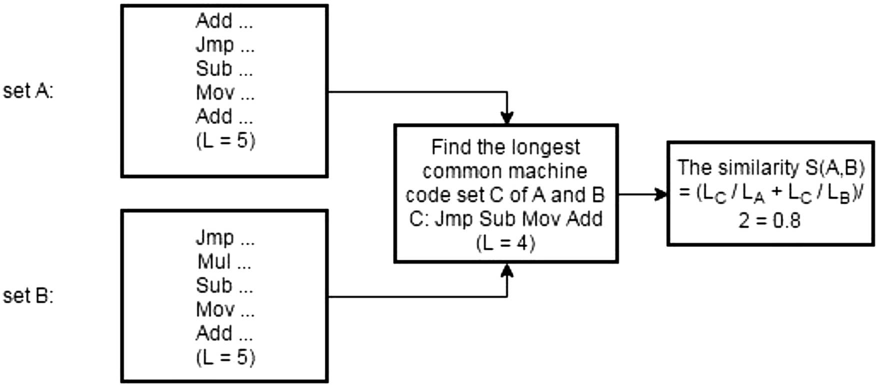


Fig. 5. Example of the algorithm of calculating the affinity of 2 set A and B

3.3 Danger Theory

3.3.1 Statistics Dangerous Levels of the Machine Code Sets

Based on Danger Theory [8], this paper proposes a method to assess the danger of a virus detector from the above training step. The method statistics frequency of virus detectors. A formula (3.3.2) is given to calculate its risk based on frequency.

3.3.2 Formula of Dangerous Calculation

After a statistic frequency of a virus detector, called % pos is the frequency of the virus files and % neg is the frequency of the benign files, the following formula is used to calculate risk DT of a virus detector as follows:

$$\mathrm{DT} = [(-\%\mathrm{neg} + \%\mathrm{pos}) + 1]/2$$

This formula is based on a logical reason that the more a detector have frequency in virus file set, the more dangerous it is, and vice versa. It adds one and divides into two for the purpose of normalization, this makes the range of DT result is always between 0 and 1. When frequency of a detector in virus file set is 1 and in benign file set is 0, the result of DT is 1. When frequency of a detector in virus file set is 0 and in benign file set is 1, the result of DT is 0. When frequency of a detector in virus file set equal to that in benign file set, the result of DT is 0.5.

4 Experiments

To evaluate the effectiveness of this approach, a program installed on C# has been made, with the different testing and training data sets.

4.1 Training and Testing Data

In this step, the virus and benign files groups with the different number was chosen to compare the test results. Each test sample was divided into two categories as the train and test to the ratio of 7:3 (Table 1).

The datasets is collected from many sources with the file extension is .exe or.com (Win32 PE file). The number of files in each dataset is from 200 to 800 files to test how that number affected the result.

Table 1. The number of files in each data sets

Datasets	Training files		Testing files	
	Virus files	Benign files	Virus files	Benign files
Dataset 1	200	150	86	64
Dataset 2	250	250	107	107
Dataset 3	400	250	171	107
Dataset 4	500	300	215	129

4.2 Experiment Results

Experimental results were shown as the following Table 2:

Table 2. The correlation between the number of files in data set and the performance of the model

Dataset	Detection Rate (%)	
	Virus files	Benign files
Dataset 1	55.3 %	92.3 %
Dataset 2	77.6 %	94.9 %
Dataset 3	90.1 %	98 %
Dataset 4	91.2 %	95 %
Average	78.55 %	95.05 %

The table showed that there was a correlation between number of training files and performance. The number of training files is very important because it increases the number of dangerous gene detector, simultaneously increasing the cover of virus malware space. The detectors were mutants with a small percentage in order to increase the ability to find out the malicious code and polymorphism hybridization, which are often used to avoid detection by antivirus programs with basic algorithms. In trials, some parameters are fixed with hand-selected threshold as minimum support value $m_{fb} = 20\ \%$ and $m_{fc} = 50\ \%$ as the best choice in multiple tests of changing data variables range, and the results showed that the average performance of 78.55 %.

5 Conclusion

This paper proposes a model combining statistical methods and machine learning algorithms inspired by biology. This combination creates the virus detectors with the ability to detect a new file is a virus or not. Experimentally shown that, if the virus samples collected more and varied, the detection of this virus can produce good results. However, some parameters of this method is not automated. In the future, this method could be improved a number of steps to increase the speed and accuracy.

Acknowledgement. This research is funded by Vietnam National University, Ho Chi Minh City (VNU-HCM) under grant number C2016-26-05.

References

1. Schultz, M.G., Eskin, E., Zadok, E., Stolfo, S.J.: Data mining methods for detection of new malicious executables, pp. 6–7 (2001)
2. Read, M., Andrews, P., Timmis, J.: Artificial immune systems, pp. 4–5 (2012)

3. Rad, B.B., Masrom, M.: Metamorphic virus variants classification using opcode frequency histogram, pp. 147–152 (2010)
4. Chao, R., Tan, Y.: A virus detection system based on artificial immune system, pp. 3–5 (2010)
5. Lu, T., Zheng, K., Fu, R., Liu, Y., Wu, B., Guo, S.: A danger theory based mobile virus detection model and its application in inhibiting virus, pp. 2–5 (2012)
6. Bilar, D.: Opcodes as predictor for malware, pp. 4–9 (2007)
7. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules, pp. 3–6 (1994)
8. Aickelin, U., Cayzer, S.: The danger theory and its application to artificial immune systems, pp. 4–6 (2008)
9. Ali, H.A., Hussain, D.J.: Computer virus detection based on artificial immunity concept, pp. 68–74 (2014)

Using Graph Database for Evidence Correlation on Android Smartphones

Minh-Tuan Dang and Tuan A. Nguyen(✉)

University of Information Technology,
Vietnam National University of Hochiminh City, Linh Trung Ward,
Thu Duc District, HCMC, Vietnam
ch1302019@gm.uit.edu.vn, tuanna@uit.edu.vn

Abstract. Nowadays, the smartphone has became the popular communication devices in everyday life. It is a main tool to facilitate the communication among people. Besides that, smartphone also save a persons activities which include many privacy information such as address book, sms messages, pictures, video, call logs, location logs, web browsing history, emails, and so much more. These documents represent the information about actions, activities or personal activities, and if there is enough lawful proof that person is guilty, that can be valuable as digital evidence in court. Therefore, mobile phone is also an important channel and analysis tool, investigate about content, data corellation which is saved in the smartphone is a valuable tool for investigator to obtain evidence from them. This paper presents the methods and technologies are used to obtain evidence from Android Smartphones. In addition, we are also point out the difficulties in both legal and technical aspects of the digital evidence obtaining from these kind of smartphones.

Keywords: Evidence correlation · Mobile forensics · Graph database

1 Introduction

The collection and analysis evidences from smartphones are crucial for investigation by providing valuable information [1,8]. However, the information collection from smartphone is still being done in a manual way by the investigators. In addition, the information that investigators can see on the smartphone is discrete information. There are a need of correlation and relation of collected information to provide evidence. Moreover, there are many types of smartphone in different formats. It is difficult to find out what information is needed for the investigation process. Because it does not shows the process of information in the chain of activities. Therefore, to find out the relation of piece of information in the mobile phone is a challenge task.

The graph database is used to store and visualise the information in the linked graph. In this paper, the authors build a graph database for collection and analysis information from Android devices to help the investigators provide

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 209–216, 2016.
DOI: 10.1007/978-3-319-48057-2_15

the sound and sense evidence by connecting pieces of information together to show the illegal acts. The input data is a collection of information collected from applications inside the Android devices that the object takes part in the illegal acts which were used as a medium for communication during the process conducting illegal acts.

In order to implement, the authors have to answer these questions: how to extract information from Android devices, and from the raw extracted information, how to organise and visualise that data into graph-database. The evidence graph-database then can be used for investigation process by using queries to discover new information. This speed up the investigation process and minimal efforts for investigators by help them view the whole picture.

The paper is organised as follow, the next section, we review related works in the area of digital mobile forensics and using graph databases for complex security scenario modeling. The third section presents the process of extracting information from Android devices. The fourth section describes the data synthesis process. The evidence graph database construction is presented in section five. Section six illustrates the evidence extraction using our modelled graph. We conclude the paper in section seven.

2 Related Works

The Android forensics is an important topic today because of the popularity of Android and it tools [6,7]. Graph database is a research has a long tradition. It is applied in various fields such as semantic web, social networks [4]. Recently accompany the development of the industry survey, graph databases are used to express the relationship between the information of evidence.

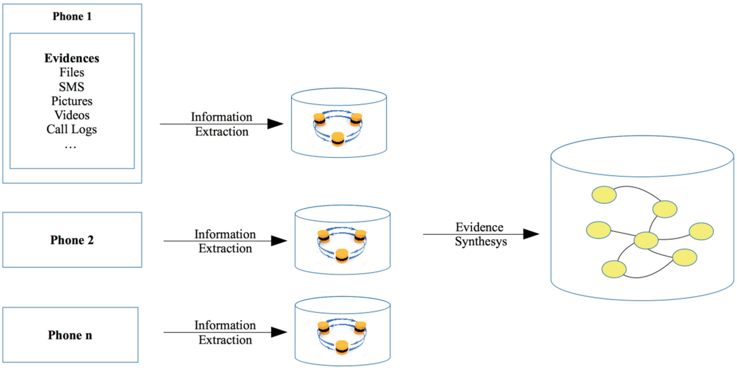


Fig. 1. The process of collecting and synthesysing evidences

The investigation of mobile devices has been conducted for a long time [3]. The process of collecting and synthesysing evidence includes the information extraction from mobile devices, then all collected data is synthersized into a graph database (See Fig. 1). In these works [1–7], the authors have presented a very deep survey and cover the main points in the field of mobile surveys. In this section, we examine three related works from these articles [8,10,11].

In [8], the authors focused on management analysis of evidence on the three social networking applications are widely used on smartphones such as Facebook, Twitter, and MySpace. The test was implemented on 3 popular smartphones: BlackBerrys, iPhones, and Android. The testing process includes the installation of social networking applications on each device examines the common behavior of users and perform the analysis of evidence on manually collected images of each set Belgium. The analysis aims to detect whether the collected behavior of these applications can be stored on the internal memory of the device or not. If yes, gas edge, the characteristics and location of the user can be found and removed from the image of the device logic. The results show that no traces were found in the BlackBerry but for iPhone and Android phone, it saves a huge amount of valuable information that can be recovered by investigators.

Recently, there is a project working on implementing a generic ICT risk model using graph databases [9]. In this work, the authors modelled the APT thread via many steps and construct a graph to represent the process of attacking. The via querying that graph, the author can find positive signals that can lead to conclude there is an APT attack happening.

In [10], the authors propose a method for automatically analyzing an online memory forensics for mobile phones. The authors have studied the behavior of the device's RAM, and the analysis is very useful in a way of the analysis of the evidence collected in real-time communications applications. Many media scenario with many different parameters are analyzed carefully. The test results show that the authors of the messages went out phone has a higher consistency is the message coming from the outside. In the authors' experiments have achieved a 100 % rate of collection of evidence from the message out. For the messages coming from the outside, the percentage collected change from 75.6 % to 100 %, by taking into account various parameters in different circumstances. Thus, in a real situation, the stakeholders can in turn send the message and continue to send several messages 1, the collection of the author can catch most of the data in support of Advanced research census.

In [11], the authors have investigated a specific example is deliberately malicious activation of telephone equipment to carry out acts of espionage. Specifically, malicious code can automatically activate the camera, from which turns sneaking microphone or tapping.

3 Information Extraction from Android Devices

All of Android applications have database storage inside the internal memory in default directory of /data/application_name/databases. However, depends on the need, application can save data in the external storage /SDCard directory. In addition, data can be saved from desktop computer without requiring root permission usually in the /SDCard. The difference between these two directories is the directory "/data/data/application_name/databases" is private. It is protected by Android, only root and the owner can access this. In contrast, the /SDCard directory is the external data storage. Data on this directory is not protected. Therefore normal users can access, edit or delete. In consequence, normal users cannot know how the database like SMS messages, address book, phone logs are stored. He/she can only view the data via their own applications.

To gain root access, the Android user needs to do rooting for the device. This can lead to providing the user the top permission. In consequence, the Android device is not protected anymore. Therefore, to protect root permission, users are recommended to install the root management application to prevent the illegal root access. The popular application for this is the Super User App. After gained the root access, the Android data extraction became simple. There are many ways to extract data, in this paper, we describe two popular methods:

– Using file manager applications and assign root permission for them. Via this application, we can save data into the SDCard and copy to desktop computers.
– Using Android debug bridge (ADB). This is a command-line tool which allows the Android and computer do communications. This tool can be found in <sdk>/platform-tools/ or download ADB driver. After installed ADB, to communicate with Android, the Android must turn on the Enabling ADB Debugging mode and assign the root permission to ADB. These two commands bellow show how to dump MMS, SMS from Android device then copy it to desktop computer:
    - adb shell su root dd if=/data/data/com.android.providers.telephony/ databases/mmssms.db of=/sdcard/mmssms.db
    - adb pull /sdcard/mmssms.db C:/WorkStarion/sms.db

The command "adb shell su root" accesses the phone using root permission; The "dd if=¡path to source files¿ of=¡path to destination file¿" command is a data copy command; the command "adb pull" is a command to pull data from device to the desktop computer. For MMS/SMS application, it has the package name is com.androidproviders.telephony and the /databases directory for saving the database which is mmssms.db. The other databases such as phone call logs, contacts, emails can be extracted in a similar manner.

The rooting device allows the investigator querying all information in Android device. However, the side effects of rooting devices can cause the device become brick or malware attack. Moreover, rooting device will result in an invalidation of the guarantee for the equipment, lastly, and more importantly will affect legitimate characters of the equipment. Because, the equipment mentioned here serves as an evidence if an investigator has the right to root and affect it too much, it will result in the problem that the evidence will lose its legitimate character. Therefore, we must think carefully before rooting the device and do it in a very carefully and thoughtful way.

Another indirect data extraction that causes fewer side effects to the device is to use a third party application for data extraction. This application is similar to the normal application, it is installed the device and request enough permission to extract data needed for investigation such as SMS reading permission, call log access permission, address book read permission. The popular application for this kind of app is the AFLogical OSE which specializes for forensics activities. This application can extract CallLogs, Contacts, and SMS into the csv format and save to /Sdcard directory. The user can use command adb pull to save from phone to desktop computer.

4 Data Synthesis from Mobile Devices

After extraction data from mobile devices, we begin to synthesise the data. In order to follow the progress of activities and their relations in a period of time, we synthesise the extracted information to become a common database (Fig. 1). There are many types of information such as SMS messages, call logs, emails, locations.

4.1 Call Logs

After successfully extracted the call logs, we collect the call logs of many mobile devices. In order to distinguish between them, we have to add each call logs its own phone number because in the call log the owner number is not represented using Linux awk command:

```
awk ‘BEGIN{FS = OFS = ‘‘,’’}
{$(NF+1) = NR==1 ? ‘‘number own’’ : ‘‘932870887 and
        963270532’’} 1’ CallLog.csv > CallLog1.csv
```

4.2 SMS Logs

Similar to Call Logs, we have to add the own number to each SMS databases. Because in SMS database, it does not show its own number. We used awk command in Linux:

```
awk ‘BEGIN{FS = OFS = ‘‘,’’} {$(NF+1) = NR==1 ? ‘‘number own’’:
   ‘‘932870887’’} 1’ SMS.csv > SMS1.csv
```

There are many different structures of SMS databases between different Android version. Therefore, we only extract information from these columns: address, date, type, body and number_own then synthesise it into a common table using awk command:

```
awk ‘FNR==1 && NR!=1{next;}{print}’ SMS*.csv > AllSMS .csv
```

5 Building Evidence Graph Database

After got the synthesis database, we convert them into a graph database to visualise it. In this work, we use Neo4J as a platform to present the database.

5.1 Grap Call Logs and SMS

In order to create a relation table for call logs, we use Neo4j command to import the data from AllCallLogs.csv into a graph database:

```
Load CSV from ''file:/home/dmtuantv/aflogical-data/
AllCallLogs.csv'' as recreate (r:relation{number1:re[1],
date:re[2],duration:re[3],type_:re[4],name1:re[6],
number2:re[9]}) return r;
```

Similarity, to create a relation table for SMS logs:

```
Load CSV from ''file:/home/dmtuantv/Phone_Data/AllSMS.csv''
as sms create (n:sms{address:sms[0],date:sms[1],type_:sms[2],
body:sms[3],number_own:sms[4]}) return n;
```

5.2 Working the Graph Database

After created a graph database, we use some Neo4J queries to refine the tables to get more useful data such as: (See Fig. 2)

- This command creates a number table which contains all numbers of the number1 column:
  match(r:relation) with distinct r.number1 as num create(n:number number:num) return n;
- This command inserts into table number include the numbers of number_own.
  match (r:sms) with distinct r.number_own as num create (n:numbersmsnumber:num) return n;
- This command creates nsms table to omit the duplicate numbers between address and number_own: *match (n:numbersms) with distinct n.number as num create (m:nsmsnumber:num) return m;*
- We can create a SEND relation from sms and nsms tables. This command creates the SEND relationship of all numbers had sent SMS:
  where n.number=r.address and n1.number=r.number_own and r.type_='1' create (n)-[:SENDdate:r.date,body:r.body] → (n1)
- This command creates SEND relationship of all numbers: *match (n:nsms), (r:sms),(n1:nsms) where n.number=r.address and n1.number =r.number_own and r.type_='2' create (n) ← [:SENDdate:r.date,body:r.body]-(n1)*

6 Evidence Graph Extraction

After the information is organized into a graph database (Fig. 3). We can start finding information on it such as:

- Find information 20/5/2015. We have to convert the date format from dd/mm/yyyy into a timestamp formart: *Match ()-[r]-() where r.date='14454.*' return r;*

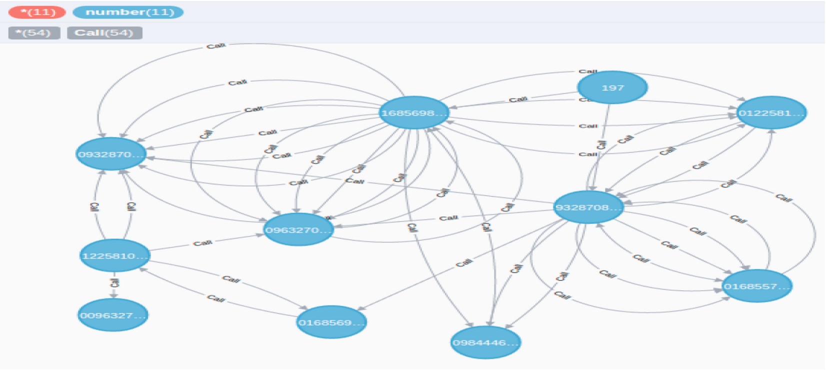
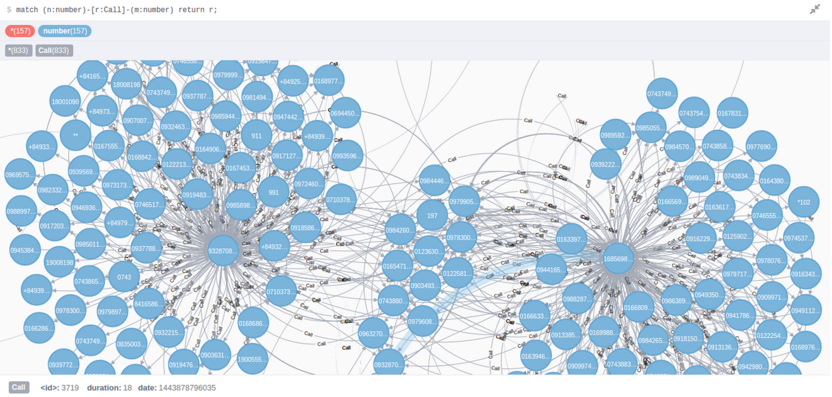


Fig. 2. Call connections

- Filter the duplicate incomming and outgoing calls: *match (n:number),(r:relation),(n1:fnumber) where n.number=r.number1 and n1.number=r.number2 and r.type_=‘3’ create (n)-[:Misseddate:r.date]-(n1) return n,n1*



```
$ match (n:number)-[r:Call]-(m:number) return r;
```

Call <id>: 3719 duration: 18 date: 1443878796035

Fig. 3. Refined duplicate calls

7 Conclusion and Future Works

This paper presents surveys and techniques for tracing evidence on Android using programming methods and evidence representation using graph database. The innovation here is that the authors have shown evidence by correlation the pieces of information together using a graph database. With the intuitive expression will contribute to the process of investigation and in particular with the professional investigators will allow them to explore more of the details related to their case.

In summary, the article “Using graph database for evidence correlation on android smartphones” has accomplished its objectives which are information extraction and presentation in graph database, as well as providing sample queries for evidence finding. We believe that the contribution of the paper will contribute to saving time investigator investigation.

The search for evidence on smartphones is becoming increasingly more difficult due to the continuous development of the operating system, the variety of mobile operating systems, the vigilance of users, particularly those intentionally crime. Moreover, the manufacturers started to enhance the system security by encrypting data of smartphones and does not provide a mechanism for decoding the police upon request. This has increased the complexity of the investigation evidence on mobile. These challenges will be an important topic in the search for evidence on mobile and will be a fascinating subject for researchers.

References

1. Angles, R., Gutierrez, C.: Survey of graph database models. ACM Comput. Surv. (CSUR) **40**(1), 11–139 (2008)
2. Barmpatsalou, K., Damopoulos, D., Kambourakis, G., Katos, V.: A critical review of 7 years of mobile device forensics. Digit. Investig. **10**(4), 323–349 (2013)
3. Brunty, J.: Mobile device forensics: threats, challenges, and future trends. In: Sammons, J. (ed.) Digital Forensics, pp. 69–84. Syngress, Burlington (2016). Chapter 5
4. Celko, J.: Graph databases. In: Celko, J. (ed.) Joe Celkos Complete Guide to NoSQL, pp. 27–46. Morgan Kaufmann, Burlington (2014). Chapter 3
5. Harichandran, V.S., Breitinger, F., Baggili, I., Marrington, A.: A cyber forensics needs analysis survey: revisiting the domain’s needs a decade later. Comput. Secur. **57**, 1–13 (2016)
6. Hoog, A.: Android Forensics: Investigation, Analysis and Mobile Security for Google Android, 1st edn. Syngress Publishing, Burlington (2011)
7. Hoog, A., Strzempka, K.: iPhone and iOS Forensics: Investigation, Analysis and Mobile Security for Apple iPhone, iPad and iOS Devices, 1st edn. Syngress Publishing, Burlington (2011)
8. Mutawa, N.A., Baggili, I., Marrington, A.: Forensic analysis of social networking applications on mobile devices. Digit. Investig. **9**(Suppl.), S24–S33 (2012). The Proceedings of the Twelfth Annual DFRWS Conference and 12th Annual Digital Forensics Research Conference
9. Schiebeck, S., Latzenhofer, M., Palensky, B., Schauer, S., Quirchmayr, G., Benesch,T., Göllner, J., Meurers, C., Mayr, I.: Implementation of a generic ICT risk model using graph databases. In: Proceedings of the Ninth International Conference on Emerging Security Information, Systems and Technologies (SECURWARE) 2015, pp. 146–153, August 2015
10. Thing, V.L.L., Ng, K.Y., Chang, E.C.: Live memory forensics of mobile phones. Digit. Investig. **7**, S74–S82 (2010)
11. Xu, N., Jia, W., Luo, Y., Zhang, F., Xuan, D., Teng, J.: An opened eye on you. IEEE Veh. Technol. Mag. **6**(4), 49–59 (2011)

Advances in Authentication and Data Access Control

DASSR: A Distributed Authentication Scheme for Secure Routing in Wireless Ad-hoc Networks

Phu H. Phung[1](✉) and Quang Tran Minh[2]

[1] Department of Computer Science, University of Dayton, Dayton, OH, USA
phu@udayton.edu
[2] Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology, VNU-HCM, Vietnam
quangtran@hcmut.edu.vn

Abstract. Secure routing is vital in wireless ad-hoc networks for establishing reliable networks and secure data transmission. However, most routing security solutions in wireless ad-hoc networks make assumptions about the availability of key management infrastructures that are against the very nature of ad-hoc networks. In this paper, we propose DASSR scheme, a new secure routing approach based on a fully distributed authentication and self-organized public key management scheme without any central authorizing entity. In DASSR, routing messages are authenticated between neighboring nodes (hop-by-hop) and between source and destination nodes (end-to-end) by using nodes' signatures. Once authenticated, messages are guaranteed for integrity and non-repudiation, hence the scheme could prevent potential routing attacks from malicious nodes. We evaluate our proposed scheme DASSR by applying it to the AODV routing protocol, a representative of reactive ad-hoc routing protocols, and demonstrate the effectiveness and security properties of the proposed approach. A comprehensive review of related secure routing protocols is presented and compared with the proposed scheme DASSR.

Keywords: Mobile · Wireless ad-hoc · Security · Secure routing protocol · Distributed · MANET

1 Introduction

A wireless ad-hoc network is a network of nodes, commonly mobile nodes, communicating to each other by self-organizing without a fixed or centralized infrastructure [7,26]. Mobile ad-hoc networks (MANETs) and wireless sensor networks (WSNs) are two instances of a wireless ad-hoc network that are widely deployed and used in practice such as in military, vehicular networks, disaster recovery [24], and many other domains. Wireless ad-hoc networks have been also integrated with Internet of Things (IoT) to carry out more powerful applications to real-world [2].

In a wireless ad-hoc network, the connectivity is ad-hoc in the sense that each node can create and join a network "on-the-fly" by performing basic networking

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 219–236, 2016.
DOI: 10.1007/978-3-319-48057-2_16

functions such as routing, forwarding, and service discovery. Nodes in a wireless ad-hoc network participate in routing processes to establish data forwarding policies for end-to-end communications. In order to realize multi-hop communications, effective multi-hop routing protocols, such as AODV [22], OLSR [25], HWMP [5], must be implemented in each node [6,31].

Ad-hoc network routing protocols mainly focus on providing the convenience for nodes to join the networks, improving collaborations between nodes in an end-to-end multi-hop communication fashion. These protocols normally assume that every node performs and follows the protocol and have not considered security aspects. However, there is no mechanism in a routing protocol to ensure that every node in an ad-hoc network follows the protocol. For example, a malicious node can modify a field in a routing message illegitimately to falsify the routing information in a network, which cannot be detected and prevented in wireless ad-hoc network standard routing protocols. This makes wireless ad-hoc networks be vulnerable with various security attacks including data forwarding attacks (e.g., denial of service, data fabrication, packet delay, data dropping and spoofing, etc.,) and network control attacks (route fabrication, making loop on networks, changing network topology) [1,3,16,29] (c.f. Sect. 2 for detailed attacks).

In addition, in the dynamic environments of ad-hoc networks, nodes are dynamically issue control messages for network establishment and management. Specifically, in the reactive routing protocol such as AODV [22], a node can issue control messages, e.g., for route request, whenever it wants to transfer data. This creates a great chance for malicious nodes to attack the network. For example, a malicious node can spoof the address of its neighbor nodes to send a false routing message to break an active routing path [1]. As a result, security for wireless ad-hoc network routing protocols is a great challenge attracting many researchers recently.

There have been a number of works proposing secure routing protocols in wireless ad-hoc networks [1,12,16,23,28,30,33,34]. Most routing security solutions such as SAODV [34], ARAN [28] assume the availability of key management infrastructures. This assumption is impractical in wireless ad-hoc networks as it violates the nature of this network architecture that does not have a fixed infrastructure. Furthermore, in our security analysis, there is no previous security scheme examining the integrity of transactions between neighbor nodes, which create security flaws for fabrication attacks. Some other works attempt to resolve this issue by proposing a cryptographic model [3] or a public-key scheme using MAC addresses on layer 2 based routing protocols [8,10]. However, the cryptographic approach needs a complex algorithm embedded in a routing protocol, while the later one cannot be able to work on layer 3 routing protocols which are commonly used in wireless ad-hoc networks.

In this work, we aim at filling the aforementioned gaps by proposing a simple yet efficient authentication scheme for secure routing based on a self-organized public-key mechanism in wireless ad-hoc networks. Our proposed scheme, namely DASSR ($\underline{\text{D}}$istributed $\underline{\text{A}}$uthentication $\underline{\text{S}}$cheme for $\underline{\text{S}}$ecure $\underline{\text{R}}$outing), is different from previous approaches that it is fully distributed without requiring a trusted server while still can defense the network against the identified attacks with low overhead. In summary, the main contributions of this paper are:

- A fully distributed authentication scheme is proposed to ensure the integrity and non-repudiation of routing messages between neighbors' nodes (hop-by-hop communications), and between the source and destination nodes (end-to-end communications) in wireless ad-hoc networks. The proposed distributed scheme is based on a key exchange and a revised self-organized public-key mechanism without a certification server.
- The proposed approach does not rely on MAC addresses for identifying public keys so that it can work on layer 3 protocols which are commonly used in ad-hoc networks. An application of the proposed scheme DASSR on AODV, a reactive routing protocol has been performed to illustrate the effectiveness of the approach.
- The proposed authentication method is based on signatures, which is simple for implementation and introduces low overhead compared to hashing or cryptography-based counterparts.
- A deep security analysis is performed to demonstrate that the proposed scheme DASSR can prevent identified attacks. We also present a comprehensive review of existing solutions and compare their security properties with DASSR.

The rest of this paper is organized as follows. Section 2 presents background of this work including routing protocols in wireless ad-hoc networks, security flaws in routing protocols, and examines current approaches to securing routing protocols. In Sect. 3, we detail our proposed distributed authentication scheme DASSR and key exchange for reactive routing protocols. We present an implementation of the proposed scheme in the AODV routing protocol in Sect. 4. Section 5 analyzes the security properties and advantages of the proposed scheme DASSR, and performs a comparison of DASSR and related secure routing schemes. We conclude our contributions and address further work in Sect. 6.

2 Background and Related Work

In this section, we review routing protocols in wireless ad-hoc networks, analyze and discuss the security issues in the existing routing protocols, and present related work.

2.1 Routing in Wireless Ad-Hoc Networks

In wireless ad-hoc networks, network topology is dynamically changed with frequent joining or moving out of mobile nodes. To realize multi-hop communications, effective multi-hop routing protocols must be implemented in each node. Two main branches of routing protocols, namely the proactive and reactive routing protocols have been proposed. The proactive protocols statically build routing tables for mobile nodes in advance (before the routes/paths are used) and periodically update those routing tables. This approach is suitable for small networks, but it is inefficient for large networks involving a huge number of control packages traveling through. Representatives of proactive routing protocols are OLSR (Optimized Link State Routing Protocol) [25], DSDV (Destination-Sequenced Distance-Vector Routing Protocol) [17], STAR (Source-Tree Adaptive Routing) [11]. In contrast, the reactive counterparts, such as AODV (Ad hoc On-Demand Distance Vector Routing Protocol) [22], DSR (Dynamic Source Routing Protocol) [13,15], DYMO (Dynamic Manet on Demand Routing Protocol) [18], TORA (Temporally Ordered Routing Algorithm) [19], examine routes on-demand when a node needs a route/path for data forwarding.

With the flexibility nature in ad-hoc networks as mobile nodes can actively issue control messages to establish routing processes for data forwarding, specifically in reactive routing protocols, the networks are exposed to attacks by malicious nodes. Typical types of attacks are described in the following sub-section.

2.2 Typical Attacks in Wireless Ad-Hoc Routing Protocols

As mentioned earlier, security issues have not been considered in ad-hoc network routing protocols. Any node can join a network, and read, forward, and send routing messages to neighbor nodes in a network without authentication. This design allows malicious nodes to launch serious attacks. In this subsection, we detail three types of attacks that are common in reactive routing protocols in wireless ad-hoc networks. The proposed scheme DASSR aims to detect and prevent these attack types.

Impersonation Attacks. A malicious node misrepresents its identity in the network so that it will break route discovery or path maintenance processes. A malicious node listens to its neighbor nodes to identify their identities and then modify its identity such as MAC or IP address in outgoing packets to generate falsified routing information: (1) a malicious node impersonates the source node, (2) a malicious node impersonates the destination node or neighbor of destination by forging a `Route Reply` with its address as a destination node, and (3) a malicious node forms a loop by spoofing nodes to change an existing route to a circle so that the message is relayed in the loop continuously without reaching the real destination.

Modification Attacks. When a malicious node is in the route discovery path, it might modify the route request or route reply. As a consequence, the discovered path causes the source node to transmit data wrong. The modification can happen for the following things: (1) route sequence numbers and (2) hop count. As for case (1), when a malicious node $\boldsymbol{M}$ receives a route request, that is destined to node $\boldsymbol{D}$ from source node $\boldsymbol{S}$, from its neighbor node $\boldsymbol{N}$, the malicious node $\boldsymbol{M}$, after re-broadcasting the message, redirects transactions toward itself by unicasting to $\boldsymbol{N}$ a `Route Reply` containing a much higher destination sequence number for $\boldsymbol{D}$ than the value last advertised by $\boldsymbol{D}$. In consequence, on receiving valid `Route Reply` from $\boldsymbol{D}$, $\boldsymbol{N}$ will discard this message. As for case

(2), malicious nodes can modify the hop count field of a `Route Request` message by resetting this value to zero or setting this value to infinity. This modification leads the route discovery process wrong.

Fabrication Attacks. In reactive routing protocols in ad-hoc network, when a node in an active path moves, the path is broken. Routing protocols such as AODV has a route maintenance mechanism to recover such broken paths. This is implemented by the node upstream of the broken link, broadcasting a `Route Error` message to all active upstream neighbors [28]. However, this mechanism is vulnerable as a malicious node may falsify an existing route by generating a `Route Error` message that in fact is not true, resulting in a denial-of-service attack in the network as nodes receiving falsified `Route Error` message cannot verify the correctness and thus delete the active path.

2.3 Related Work

In the following paragraphs we summarize related work for security in wireless ad-hoc networks.

Data Forwarding Security. Several works have been dedicated for data forwarding security in distributed systems like wireless sensor networks and MANETs, where there is no centralized element to manage the security policy. Rezvani *et al.*, proposed a collaborative-based reputation method to which the credibility of each node is evaluated by other nodes in the network [27]. Based on the credibility, the trustworthiness of mobile nodes is measured. This allows the network to detect malicious or untrusted nodes, protecting network nodes from receiving data from attackers. The accuracy of the propagated credibility is validated using the variances of sensors whereby the distribution of noise in sensors is modeled by Gaussian distribution which is not always correct in the real wireless environments. In addition, the credibility propagation may also include judgments from untrusted/malicious nodes.

Routing Security. Beside data forwarding security, because of its nature, wireless ad-hoc network is significantly vulnerable with routing security as routing establishment and management are essential and these processes are conducted frequently. Various researches have been dedicated for routing security methods which mainly rely on cryptography [3]. Ben-Othman *et al.*, proposed an Identity Based Cryptography (IBC) method for node identity in the Hybrid Wireless Mesh Protocol (HWMP) [8–10] for IEEE 802.11s mesh network [5]. As HWMP is a layer 2 routing protocol, MAC addresses of mobile nodes are used as the public keys for the control messages such as route request (RREQ), route reply (RREP). As a result, this approach does not need a centralized entity to verify the authentication of public keys. Therefore, it is suitable for security routing in infrastructure-less ad-hoc networks. The essential issues in this method, however, are that (i) IEEE 802.11s is not the only standard protocol for ad-hoc networks, meanwhile more commonly used routing protocols work on layer 3 where IP address is used instead of MAC address; (ii) theoretically, MAC address can also be faked by malicious nodes thus the system needs a secured scheme to protect MAC address fabrication. Our work in this paper is different which focuses on layer 3 secure routing protocols which are commonly used in ad-hoc networks.

In another aspect, there have been a number of solutions for securing routing protocols working on layer 3 in wireless ad-hoc networks such as [1,4,12,16,20,23,28,30,33,34]. These solutions both have advantages and disadvantages. The most common disadvantage is that they assume a fixed infrastructure, which is against the nature of ad-hoc networks, and is complex to implement in practice. In a previous work [23], we proposed a hash-based authentication scheme among two nodes to authenticate the messages without introducing a fixed infrastructure. While that approach can ensure the integrity of messages, it still be open to fabrication attacks as there is no end-to-end authentication between the source and destination nodes. The proposed scheme DASSR in our work overcomes these weaknesses by introducing a fully distributed authentication mechanism without a fixed server while providing end-to-end authentication. We present DASSR in detail in the next section and perform a comprehensive comparison of the proposed scheme DASSR and related solutions in Sect. 5.2.

3 Distributed Authentication Scheme for Reactive Routing Protocols

Reactive routing protocols demonstrate the effectiveness in wireless ad-hoc networks as it works on-demand, reducing the broadcasting messages for updates. However, this feature makes the network vulnerable to attacks since there is no mechanism to authenticate the messages from neighbor and source nodes. Without authentication, reactive routing protocols are vulnerable to three main attack categories: impersonation attacks, modification attacks, and fabrication attacks as analyzed in Sect. 2. Our approach to preventing these potential attacks is to authenticate all messages in a routing protocol. Using authentication, routing messages are guaranteed two main properties:

Integrity. This property ensures that the content of routing messages from an untrusted node cannot be altered or modified by malicious/unauthorized nodes thanks to the signature verification. The integrity of routing messages are guaranteed by a hop-to-hop and end-to-end authentication mechanism.

Non-repudiation. Routing messages are signed using a private key by the sending node, and will be validated by the receiver using public key of the sender. The successful validation guarantees the non-repudiation of the messages, which ensures that the messages are sent by the node signed the messages and cannot be spoofed by other nodes.

The authentication process in DASSR is performed in 2 steps: hop-by-hop authentication at intermediate nodes and end-to-end authentication at the destination node. The authentication uses RSA Public-key crypto-system: messages will be signed by the sender using its private key and verified by the receiver using the sender's public key. The original message with signature from a sender will be forwarded to the destination receiver so that the receiver can verify its integrity. Thus in this scheme, each node needs to store a public-key repository for the authentication process. In the following subsections, we present the process in detail.

3.1 Overview of the Proposed Scheme

The overview of the proposed scheme DASSR is depicted in Fig. 1 and explained as follow. Before a source node $\boldsymbol{S}$ sends/broadcasts a routing message $\mathcal{M}$ according to a routing protocol, it first signs $\mathcal{M}$ with its private key to create a signature *signature_S* and attach to $\mathcal{M}$. Then $\boldsymbol{S}$ broadcasts the signed message [$\mathcal{M}$,*signature_S*]. When its neighbor node $\boldsymbol{n}$ receives the signed message, $\boldsymbol{n}$ uses the public key of $\boldsymbol{S}$ to verify *signature_S*. If the verification succeeds and $\boldsymbol{n}$ is not the destination, it additional signs the message with its private key then forwards the double signed message [$\mathcal{M}$,*signature_S*,*signature_n*] further. At any intermediate node $\boldsymbol{i}$, once it receives a double signed message, it verifies the second signature (which is signed by a neighbor node). The verification ensures the integrity of the message from the neighbor node. If the destination address does not match with $\boldsymbol{i}$'s address, it signs the message and generates its signature *signature_i*, then replaces the *signature_n* by its own signature *signature_i*, and finally forwards the new double signed message further. This process is similar at a destination node $\boldsymbol{d}$ except when checking if the destination address matches with $\boldsymbol{d}$'s address, $\boldsymbol{d}$ needs to verify the signature of $\boldsymbol{S}$ *signature_S*.

In summary, routing messages are authenticated among intermediate nodes (hop-by-hop) and from the source to destination nodes (end-to-end authentication). The authentication is based on signature using RSA Public-key Cryptosystem [14]. Thus, each node in a network generates its own pair of public key $\boldsymbol{PuK}$ and private key $\boldsymbol{PrK}$. For hop-by-hop authentication, each node keeps track of a list of neighbor nodes and for each neighbor, maintains its neighbors'

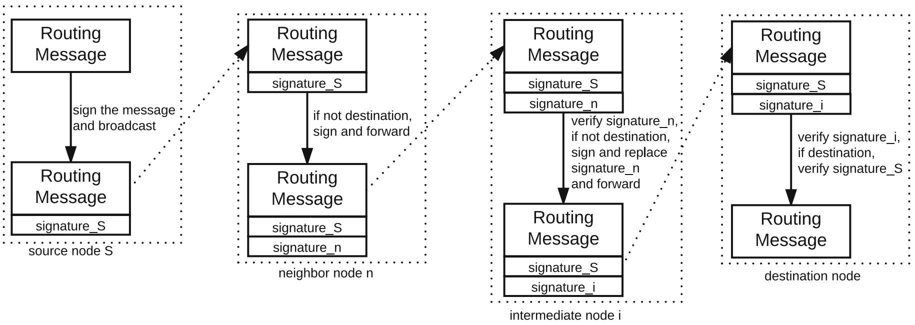


Fig. 1. Overview of the proposed distributed authentication scheme DASSR for secure routing.

address and public key. For end-to-end authentication, a destination node needs to keep the public key of the sender in order to verify the signature. The big challenge in this scheme is how each node can keep public keys of other nodes for the authentication. Using or assuming a centralized element or certification server to distribute public keys or certifications is not suitable for wireless ad-hoc networks. Our approach for this issue is that each node when joining the network first exchange its public key with the neighbor nodes. Each node in the network exchanges its public key repository to neighbor nodes so that eventually, any node in the network will have public keys of the other nodes in a self-organized and distributed manner without having a central element. These steps are presented in detail below.

3.2 Public Key Exchange Process Between Neighbor Nodes

A node that wants to join a network sends a join request message to its neighbors to exchange their `public keys`. The node broadcasts a message requesting the key exchange to its neighbor nodes. The receiving node responds with a join reply message that includes its public key. The pseudo-code algorithm is given in Fig. 2.

Sender i:

```
1. Generate RSAPairKey = (PuK, PrK);
2. Broadcast Join.Req = (AGREEMENT_REQ, request_id, sender_addr, PuK);
```

Receiver j:

```
1. Receive a message;
2. If packetType == AGREEMENT_REQ then
   Send Join.Rep =
       (AGREEMENT_REP, request_id, sender_addr, neighbor_addr, PuK);
Else if packetType == AGREEMENT_REP then
   Store it to public key list;
```

Fig. 2. Public key exchange process between neighbor nodes

A sender ***i*** before joining the network first generates its RSA private and public key pair ***PuK, PrK***, then it broadcasts a join message with the packet type ***AGREEMENT_REQ***, together with the request id, its address and public key ***PuK***.

Once a node receives a message with a packet type ***AGREEMENT_REQ***, it will unicast back the key exchange agreement message with the packet type ***AGREEMENT_REP***, together with the request id, its address, and the neighbor address and its public key ***PuK***. If the packet type is ***AGREEMENT_REP***, it will extract the neighbor public key and store in its neighbors' public key list.

3.3 Public Key and Certificate Repository Exchange

In our proposed scheme DASSR, an end-to-end authentication must be performed to ensure the integrity and non-repudiation of the original message from the source node. Therefore, a node must keep the public key of other nodes in the network to be able to verify the signature of the source node. To this end, we adopt and revise self-organized public-key management scheme proposed by Capkun *et al.,* [32], which is suitable for wireless ad-hoc networks as the management scheme does not rely on any trusted authority or fixed server.

Overview of Self-organized Public-Key Management Scheme [32]. This public key management scheme works based on the following principle:

> *"If a user **u** believes that a given public key **PuK_v** belongs to a given user **v**, the user **u** can issue a public-key certificate in which **PuK_v** is bound to **v** by the signature of **u**."*

Based on that principle, nodes that receive the newly issued certificate from a neighbor add it to their own certificate graph and further distribute the updated certificate graph. When a node ***u*** wants to verify the authenticity of another node ***v***'s public key, it merges its local certificate repositories and then evaluates the authenticity of ***PuK_v*** from the merged repository.

The scheme also provides the way that detects misbehaving users and resolves the conflicting certificates during operation. The solution requires users conscious involvement in creating their public/private key pairs and issuing certificates; all other operations (including certificate exchange and construction of certificate repositories) are fully automatic.

Revised Scheme for Public Key and Certificate Repository Exchange. We adopt and revise the aforementioned self-organized public key management scheme to apply in our distributed authentication scheme to ensure end-to-end authentication. The modified scheme is detailed below.

Nodes exchange their certificate graph and construct their updated certificate repository by following the certificate exchange process given in [32]. The scheme is applied to our scheme with a modification in which after certificate repository exchange, nodes perform public key exchange with its neighbors.

Upon receiving a public key exchange request, a node validates the public key by looking up its certificate repository. If found, it exchanges the public key and then store the tuple (node id, public key) in its trusted neighbor list. If not found, it waits for the convergence time T_{CE} (that is the expected time after which, when issued, a certificate reaches all the nodes in the network [32]) and then looks up its latest updated certificate graph again. If still not found, it refuses to exchange public key.

Through this exchange process, a node will have a up-to-date certificate repository and the list of trusted neighbors' public-keys. The certificate graph is used for authentication of end-to-end transactions in a network; whereas the list of neighbors public-key is used for hop-by-hop authentication.

According to this scheme, at least one node in the network has to issue the certificate for inclusion of a new node. If the certificate is issued, its public-key certificate is distributed throughout the network during the convergence time. Otherwise, it means that no node in the network know the new node. Thus, the new node is not allowed to take part in the further networking activities.

4 An Implementation of the Proposed Scheme DASSR for the AODV Routing Protocol

To demonstrate how our proposed authentication scheme works in a particular reactive routing protocol, we deploy the scheme in the AODV (Ad hoc On-Demand Distance Vector) routing protocol. In this section, we first review the AODV protocol, then we present a secure AODV protocol using our distributed authentication scheme DASSR.

4.1 AODV Protocol

As mentioned earlier, AODV protocol is a routing protocol for wireless ad-hoc networks using a reactive routing approach, which does not keep every node in the network on a routing table but builds a path on-demand. The routing protocol has three main different packets: Route Request ($\mathcal{RREQ}$), Route Reply ($\mathcal{RREP}$), and Route Error ($\mathcal{RERR}$) [21,22]. When a node wants to send a message to a destination that is not cached in the routing table, it issues a Route Request ($\mathcal{RREQ}$). When a node receives a Route Request, it forwards further or issues a Route Reply if it is the destination node or it has a fresh-enough route to the destination. When a node issues a Route Reply, it constructs $\mathcal{RREP}$ message and unicasts back to the neighbor node in the reverse path. A Route Error message will be issued and broadcasted when there is an error in a discovered path.

Similar to other routing protocols in wireless ad-hoc networks, AODV was designed without security consideration. Hence it is also vulnerable to typical attacks such as impersonation, modification, fabrication attacks presented in Sect. 2.2. In the next subsection, we present the implementation of our distributed authentication scheme DASSR in AODV to secure the routing protocol.

4.2 Secure AODV Using the Proposed Distributed Authentication Scheme DASSR

In the following revised AODV protocol, we assume that the public key exchange process presented in the previous section have been performed and completed. Thus, each node has neighbor nodes' public keys and a certificate repository of other active nodes in the network.

Route Request. A node having a packet to send, so-called a source node ***S***, initiates a $\mathcal{RREQ}$ message. Eventually, this message arrives at the destination node through the forwarding of zero or more intermediate nodes. In our scheme DASSR, the source node ***S*** attaches its signature signed with its private key ***PrK_S*** to the $\mathcal{RREQ}$ message as follows:

```
Message = (RREQ, signature_S)

where
signature_S = [bcastId, destAddr, destSeq, srcAddr, srcSeq]PrK_S
```

Note that the signature *signature_S = [bcastId, destAddr, destSeq, srcAddr, srcSeq]PrK_S* indicates that it is signed by the private key ***PrK_S*** on the content in [..], which are broadcast ID, destination address, destination sequence number, source address, source sequence number. These are non-mutable fields in a $\mathcal{RREQ}$ message. The content is not encrypted so that any receiving node can read to perform the routing protocol.

On receiving $\mathcal{RREQ}$ with a single signature from the source node ***S***, a neighbor node ***n*** first assures the integrity of the message by validating the source's signature with the source's public key. If the message is valid, the node continues the steps in the AODV routing protocol such as updating the hop count in the $\mathcal{RREQ}$.

The neighbor node ***n*** generates its own signature and appends this signature to the message before forwarding. The new message contains:

```
Message = (RREQ, signature_S, signature_n)
where
signature_n = [bcastId, destAddr, destSeq, srcAddr, srcSeq, hopcount]PrK_n
```

Continuing with this revised AODV protocol, any intermediate node ***i*** (except the neighbor node ***n*** which is one-hop from the source node) will receive a double signed $\mathcal{RREQ}$ message. Upon receiving the double signed $\mathcal{RREQ}$ message, node ***i*** validates the signature of the forwarding node only. This authentication process follows a hop-by-hop authentication that uses the exchanged and trusted public-keys. If the validation succeeds, node ***i*** signs the $\mathcal{RREQ}$ message similar to node ***n*** above, and replaces the forwarding node's signature with its own signature, and then rebroadcasts the message:

```
Message = (RREQ, signature_S, signature_i)
```

The signatures of intermediate nodes help preventing spoofing attacks. In this way, the authentication process is performed in a hop-by-hop manner based on the list of trusted neighbors' public-keys, without accessing the local certificate repository.

Repeating this procedure, the authenticated $\mathcal{RREQ}$ message arrives at the destination node. Note that all intermediate nodes do not validate the signature of the source node ***S*** in our scheme, except the neighbor node ***n*** and the destination node or an intermediate node initiating a `Route Reply`. At the destination, it first validates the forwarding (neighbor) node's signature and then validates the signature of the source node *signature_S*. If the validation succeeds, the message is ensured its integrity (content is unaltered by unauthorized nodes) and non-repudiation (it was actually sent by the source node $\boldsymbol{S}$). The destination node validates the signature of its neighbor node as the same procedure as that in the intermediate nodes presented in the previous section.

For the signature of the source node, after successfully authenticating the neighbor nodes signature, the destination node verifies the signature of the source node to authenticate the original route request from the source node. In our public key exchange mechanism presented in Sect. 3.3, the destination node should have the public key of the source node. Thus, the destination node of a $\mathcal{RREQ}$ can validate the signature of the source node. If the validation is successful, the $\mathcal{RREQ}$ message is guaranteed the integrity and non-repudiation from source node to destination node and among intermediate nodes. This end-to-end authentication process can prevent the modification attacks and impersonation attacks that cannot been solved in a hop-by-hop authentication method such as in [23].

In the case that the destination node cannot find the public-key of the source node in its repository, it still can apply the authentication process proposed in [32]. According to [32], when a user $\boldsymbol{u}$ wants to authenticate a public key $\boldsymbol{PuK\_v}$ of another user $\boldsymbol{v}$, both nodes merge their updated certificate repositories and $\boldsymbol{u}$ tries to find a certificate chain to $\boldsymbol{v}$ in the merged repository. If the certificates are both valid and correct, $\boldsymbol{u}$ authenticates $\boldsymbol{PuK\_v}$. Here again, $\boldsymbol{u}$ performs the certificate correctness check locally. If node $\boldsymbol{u}$ cannot find any certificate chain to $\boldsymbol{PuK\_v}$, it aborts the authentication.

Route Reply. In the AODV routing protocol, a route reply message ($\mathcal{RREP}$) is initiated by either the destination or intermediate nodes which have a fresh-enough route to the destination.

In this secure AODV protocol, the node initiating a route reply $\mathcal{RREP}$ message signs the message by its own private key and unicasts back to the neighbor node in the reverse path. The neighbor node of the initiating node validates the signature of source node (physical neighbor) and then attaches its signature to the message and forwards back to the next hop in the reverse path. Each node along the reverse path back to the source, on receiving the $\mathcal{RREP}$ message, validates the signature of the senders by using their trusted neighbors public key list, replaces the signature of neighbor node by its own one and forwards back to the next hop. When the source node receives the $\mathcal{RREP}$ message, it validates the two signatures. This process is similar to the destination node validates the $\mathcal{RREQ}$ message presented previously.

Route Error. `Route Error` ($\mathcal{RERR}$) message in the route maintenance process is another target for attacks; hence, it needs to be authenticated. The procedure for authentication of route error is the same as $\mathcal{RREP}$ authentication process.

In route reply and route error processes, if all validations succeed, the $\mathcal{RREP}$ message is guaranteed the integrity and non-repudiation for end-to-end transactions; therefore, this solution could prevent possible attacks mentioned above.

5 Evaluation and Comparison

5.1 Security Analysis of the Proposed Scheme DASSR

As presented and discussed above, our proposed authentication scheme DASSR is fully distributed without any fixed server. Since the validation of signatures does not need any central server, the authentication process imposes less overhead on the network because it does not need to communicate with a server for verification. In addition, the messages themselves are not encrypted, thus reduce the computation overhead at nodes.

As discussed in Sect. 3, using the message authentication in DASSR scheme, the integrity and non-repudiation of routing messages are guaranteed among nodes that include source, destination, and intermediate nodes. The integrity of messages ensures that the content of messages is unaltered by a malicious node. The non-repudiation guarantees that a received message came from the node did construct and sent the message, a malicious node cannot spoof another node to send a message thanks to signature verification. Therefore, our DASSR scheme can prevent potential routing attacks including impersonation, fabrication, and modification attacks. We present the detailed analysis as follows.

Impersonation attacks: By using hop-by-hop and end-to-end signature validation, our DASSR scheme can prevent any malicious node from spoofing the MAC or IP address of other nodes. If a malicious node constructs a falsified routing message using a spoofed address, the signature validation is failed because the address does not match with its public key thanks to the signature. If the signature validation is failed, the received messages are dropped.

Fabrication attacks: Any malicious node can generate a wrong *route error* message to falsify the network. However, our DASSR scheme authenticates any type of message in the network. Therefore, malicious nodes cannot spoof other nodes address to falsify route errors. Nevertheless, any trusted node can initiate wrong information to do the network harm. Since the scheme ensures the non-repudiation of messages, a trusted node that continues to inject false messages into the network can be detected and thus deleted from trusted list of neighbors, being excluded from future routing activities.

Modification attacks: Modifications such as source ID or destination sequence number are detected by the end-to-end authentication. However, the falsified modification of hop-count field can not be detected. For this case, we just rely on the transitively trusted relationship in which all nodes in the network are trusted directly or indirectly via some other nodes.

5.2 Comparison

In this subsection, we review the state-of-the-art on secure *reactive* routing protocols in wireless ad-hoc networks and compare our DASSR scheme with the existing secure routing protocols in literature.

ARAN [28]. ARAN (Authenticated Routing for Ad-hoc Networks) uses cryptographic certificates to achieve authentication, message integrity and non-repudiation in the route discovery process. It assumes the existence of a trusted certificate server which forms a center element.

SAODV [34]. SAODV (Secure Ad-hoc On-Demand Vector) routing protocol guarantees security based on a key management scheme in which each node must have certificated public keys of all nodes in the network. This protocol uses public key distribution approach. Therefore, it is difficult to deploy and it costs high since it requires both asymmetric cryptography and hash chains in exchanging messages.

OSR [4]. OSR, stands for On-demand Secure Routing Protocol Resilient to Byzantine Failures, floods route request and reply messages to prevent Byzantine failures. It uses digital signature to authenticate the source, however it requires a public key infrastructure.

Ariadne [12]. Ariadne, stands for Secure On-Demand Routing Protocol, provides point-to-point authentication of routing messages using MAC (Message Authentication Code) based on a shared key between two nodes. It assumes that sender and receiver establish the shared key before exchanging routing messages.

IBC [10]. IBC (Identity Based Cryptography) uses MAC addresses and cryptography to secure routing messages. MAC addresses are used as public keys, therefore, the mechanism does not require a centralized entity. However, as discussed earlier, this protocol is applicable for layer 2 routing protocols while more commonly used routing protocols work on layer 3 where IP address is used.

ESARP [33]. ESARP (Efficient Security Aware Routing Protocol) uses an asymmetric encryption to encrypt routing messages. It uses a key exchange scheme to distribute public keys so that it does not require a centralized server. However, the encryption introduce high overhead in computation.

In summary, existing schemes for secure routing are either based on the assumptions of the availability of key management infrastructures, which are against the very nature of ad-hoc networks [4,12,28,34], or not applicable for every ad-hoc network protocols [10], or high overhead due to complex cryptographic algorithms [10,33]. Our scheme DASSR authenticates routing messages

Table 1. Comparison of secure routing protocols with DASSR.

Scheme	Security	Verification mechanism	Fixed infrastructure required
ARAN [28]	Encryption	Public Key Cryptography	Trusted certificate server
SAODV [34]	Authentication	Digital Signature	Key Distribution System
OSR [4]	Authentication	Digital Signature	Public Key Infrastructure
Ariadne [12]	Authentication	MAC[a]	Key Distribution Center
IBC [10]	Encryption	Cryptography	None
ESARP [33]	Encryption	Cryptography	None
DASSR	Authentication	Digital Signature	None

[a] Message Authentication Code.

using digital signature without encryption, therefore it creates less overhead. The public key exchange in DASSR is fully distributed without any centralized element or fixed infrastructure. DASSR can prevent identified routing attacks in ad-hoc networks as analyzed in Sect. 5.1. Table 1 shows a comprehensive comparison of our DASSR scheme compared with existing secure routing solutions.

6 Conclusion and Future Work

In this work, we proposed DASSR, a fully distributed hop-by-hop and end-to-end authentication scheme for reactive routing protocols in wireless ad-hoc networks. Its advantages are two-fold: (1) It uses an efficient hop-by-hop authentication scheme, which prevents impersonation, modification, and fabrication attacks, during path discovery without resorting to any central entity. (2) The proposed scheme also provides an end-to-end authentication mechanism by adapting a self-organized public-key management scheme. In this way, our DASSR scheme can ensure the integrity and non-repudiation of original messages from the source node, thus can prevent modification attacks without relying on a certificate server. Our scheme can work on layer 3 protocols (as it does not rely on MAC addresses) which are widely used in wireless ad-hoc networks. We demonstrate the security properties and the effectiveness of the proposed scheme by deploying it to the AODV protocol, a representative of reactive ad-hoc network routing protocols. Secure routing protocols adopted the proposed scheme DASSR do not use cryptography or rely on a central server, therefore the overhead is low. In the future work, we will implement the DASSR scheme in other reactive routing protocols and compare its overhead and performance with other related secure routing protocols to confirm the effectiveness as well as the efficiency of the proposed approach.

Acknowledgments. This material is based on research sponsored partially by grants from University of Dayton Research Council and the Swedish Research Council (through Chalmers University of Technology). The authors would also like to thank the three anonymous reviewers for their helpful feedbacks.

References

1. Abdelaziz, A.K., Nafaa, M., Salim, G.: Survey of routing attacks and countermeasures in mobile ad hoc networks. In: 15th International Conference on Computer Modelling and Simulation (UKSim 2013), pp. 693–698, April 2013
2. Alcaraz, C., Najera, P., Lopez, J., Roman, R.: Wireless sensor networks and the internet of things: do we need a complete integration. In: 1st International Workshop on the Security of the Internet of Things (SecIoT 2010). IEEE, December 2010
3. Andel, T.R., Yasinsac, A.: Surveying security analysis techniques in MANET routing protocols. IEEE Commun. Surv. Tutorials **9**(4), 70–84 (2007)
4. Awerbuch, B., Holmer, D., Nita-Rotaru, C., Rubens, H.: An on-demand secure routing protocol resilient to byzantine failures. In: Proceedings of the 1st ACM Workshop on Wireless Security, WiSE 2002, NY, USA, pp. 21–30 (2002). http://doi.acm.org/10.1145/570681.570684
5. Bahr, M.: Proposed routing for IEEE 802.11s WLAN mesh networks. In: Proceedings of the 2nd Annual International Workshop on Wireless Internet, WICON 2006, NY, USA (2006). http://doi.acm.org/10.1145/1234161.1234166
6. Bakht, H.: Survey of routing protocols for mobile ad-hoc network. Int. J. Inf. Commun. Technol. Res. **1**(6), 258–270 (2011)
7. Baryun, A., Al-Begain, K., Villa, D.: A hybrid network protocol for disaster scenarios. In: Fifth IEEE International Conference on Next Generation Mobile Applications, Services and Technologies, pp. 129–136, September 2011
8. Ben-Othman, J., Benitez, Y.I.S.: On securing HWMP using IBC. In: 2011 IEEE International Conference on Communications (ICC), pp. 1–5, June 2011
9. Ben-Othman, J., Mokdad, L., Benitez, Y.I.S.: Performance comparison between IBC-HWMP and Hash-HWMP. In: Global Telecommunications Conference (GLOBECOM 2011), pp. 1–5. IEEE, December 2011
10. Ben-Othman, J., Saavedra Benitez, Y.I.: IBC-HWMP: a novel secure identity-based cryptography-based scheme for hybrid wireless mesh protocol for IEEE 802.11s. Concurr. Comput. Pract. Exp. **25**(5), 686–700 (2013). http://dx.doi.org/10.1002/cpe.1813
11. Garcia-Luna-Aceves, J.J., Spohn, M.: Source-tree routing in wireless networks. In: Seventh International Conference on Network Protocols (ICNP 1999), pp. 273–282, October 1999
12. Hu, Y.C., Perrig, A., Johnson, D.B.: Ariadne: a secure on-demand routing protocol for ad hoc networks. Wirel. Netw. **11**(1–2), 21–38 (2005). http://dx.doi.org/10.1007/s11276-004-4744-y
13. Johnson, D.B.: Routing in ad hoc networks of mobile hosts. In: The Workshop on Mobile Computing Systems and Applications, pp. 158–163. IEEE Computer Society (1994)
14. Jonsson, J., Kaliski, B.: Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1 (2003)

15. Kanthe, A.M., Simunic, D., Prasad, R.: Comparison of AODV and DSR on-demand routing protocols in mobile ad hoc networks. In: 1st International Conference on Emerging Technology Trends in Electronics, Communication and Networking (ET2ECN 2012), pp. 1–5, December 2012
16. Karlof, C., Wagner, D.: Secure routing in wireless sensor networks: attacks and countermeasures. Ad Hoc Netw. **1**, 293–315 (2003)
17. Khan, K.U.R., Zaman, R.U., Reddy, A.V., Reddy, K.A., Harsha, T.S.: An efficient DSDV routing protocol for wireless mobile ad hoc networks and its performance comparison. In: Second UKSIM European Symposium on Computer Modeling and Simulation (EMS 2008), pp. 506–511, September 2008
18. Kum, D.W., Park, J.S., Cho, Y.Z., Cheon, B.Y.: Performance evaluation of AODV and DYMO routing protocols in MANET. In: 7th IEEE Consumer Communications and Networking Conference (CCNC 2010), pp. 1–2, January 2010
19. Kuppusamy, P., Thirunavukkarasu, K., Kalaavathi, B.: A study and comparison of OLSR, AODV and TORA routing protocols in ad hoc networks. In: 3rd International Conference on Electronics Computer Technology (ICECT 2011), vol. 5, pp. 143–147, April 2011
20. Lee, Y.H., Kim, H., Chung, B., Lee, J., Yoon, H.: On-demand secure routing protocol for ad hoc network using ID based cryptosystem. In: Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies, PDCAT 2003, pp. 211–215, August 2003
21. Perkins, C., Belding-Royer, E., Das, S.: Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561 (Experimental) July 2003. http://www.ietf.org/rfc/rfc3561.txt
22. Perkins, C.E., Royer, E.M.: Ad-hoc on-demand distance vector routing. In: Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 1999), pp. 90–100, February 1999
23. Phu, P.H., Yi, M., Kim, M.-K.: Securing AODV routing protocol in mobile ad-hoc networks. In: Hutchison, D., Denazis, S., Lefevre, L., Minden, G.J. (eds.) IWAN 2005. LNCS, vol. 4388, pp. 182–187. Springer, Heidelberg (2009). doi:10.1007/978-3-642-00972-3_15
24. Quang, T.M., Yoshitaka, S., Cristian, B., Shigeki, Y.: On-site configuration of disaster recovery access networks made easy. Ad Hoc Netw. **40**, 46–60 (2016). Elsevier
25. Rousseau, S., Benbadis, F., Lavaux, D., San L.: Overview and optimization of flooding techniques in OLSR. In: WoWMoM 2011, pp. 1–7, June 2011
26. Ray, N.K., Turuk, A.K.: A framework for disaster management using wireless ad hoc networks. In: Proceedings of the 2011 International Conference on Communication, Computing and Security, ICCCS 2011, NY, USA, pp. 138–141 (2011). http://doi.acm.org/10.1145/1947940.1947970
27. Rezvani, M., Ignjatovic, A., Bertino, E., Jha, S.: A collaborative reputation system based on credibility propagation in WSNs. In: IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS 2015), pp. 1–8, December 2015
28. Sanzgiri, K., LaFlamme, D., Dahill, B., Levine, B.N., Shields, C., Belding-Royer, E.M.: Authenticated routing for ad hoc networks. IEEE J. Sel. Areas Commun. **23**(3), 598–610 (2005)
29. Sen, J.: Routing security issues in wireless sensor networks: attacks and defenses. In: Sustainable Wireless Sensor Networks (2011)
30. Sivakumar, M., Jayanthi, M.K.: Reliability analysis of link stability in secured routing protocols for MANETs. Eng. J. **18**, 66–76 (2014)

31. Taneja, S., Kush, A.: A survey of routing protocols in mobile ad hoc networks. Int. J. Innov. Manag. Technol. **1**(3), 279–285 (2010)
32. Čapkun, S., Buttyán, L., Hubaux, J.P.: Self-organized public-key management for mobile ad hoc networks. IEEE Trans. Mobile Comput. **2**(1), 52–64 (2003). http://dx.doi.org/10.1109/TMC.2003.1195151
33. VinothKumar, K., Rajaram, A.: An efficient security aware routing protocol for mobile ad hoc networks. Int. J. Comput. Sci. Netw. Secur. **14**(12), 66–73 (2014)
34. Zapata, M.G.: Secure ad hoc on-demand distance vector routing. SIGMOBILE Mob. Comput. Commun. Rev. **6**(3), 106–107 (2002). http://doi.acm.org/10.1145/581291.581312

A Secure Token-Based Communication for Authentication and Authorization Servers

Jan Kubovy[2], Christian Huber[1](✉), Markus Jäger[1], and Josef Küng[1]

[1] Institute for Application Oriented Knowledge Processing (FAW),
Faculty of Engineering and Natural Sciences (TNF),
Johannes Kepler University (JKU), Linz, Austria
{chuber,mjaeger,jkueng}@faw.jku.at
[2] Informations- u. Prozesstechnik, Anwendungen,
Eigenentwicklungen, Stadtwerke München GmbH, München, Germany
kubovy.jan@swm.de
http://www.faw.jku.at, https://www.swm.de

Abstract. Today, software projects often have several independent subsystems which provide resources to clients. To protect all subsystems from unauthorized access, the mechanisms proposed in the OAuth2.0 framework and the OpenID Standard are often used. The communication between the servers, described in the OAuth2.0 framework, must be encrypted. Usually, this is achieved using Transport Layer Security (TLS), but administrators can forget to activate this protocol in the server configuration. This makes the whole system vulnerable. Neither the developer, nor the user of the system is able to check whether the communication between servers is safe. This paper presents a way to ensure secure communication between authentication-, authorization-, and resource servers without relying in on a correct server configuration. For this purpose, this paper introduces an additional encryption of the transmitted tokens to secure the transmission independently from the server configuration. Further this paper introduces the Central Authentication & Authorization System (CAAS), an implementation of the OpenId standard and the OAuth2.0 framework that uses the token encryption presented in this paper.

Keywords: OpenID · OAuth2.0 · Security · Authentication · Authorization · Token · Encryption

1 Introduction

This work is based on and partially implements *OpenID* [1] and *OAuth2.0* [2]. It presents a cloud environment with several independent actors, clients, resource-, authorization- and authentication servers. Within the system, a user can access data and services from resource servers. To obtain access, the user has to prove his identity to the authentication server. Further, the authorization server has to confirm that this specific user has permission to access the resource.

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 237–250, 2016.
DOI: 10.1007/978-3-319-48057-2_17

The OAuth2.0 framework requires secure communication between the involved parties. Otherwise an attacker could gain access to secured resources. To prevent this, a cryptographic protocol has to be enabled by the server administrator, explicitly [2,3]. According to the *SSL Pulse* project, most Internet sites use no or inadequate security protocols [4]. A user may verify that the connection of his client is secure, but he can not check whether the communication between the servers is secure. Therefore, our approach introduces an additional encryption of the sensitive information on the application level.

This paper starts by discussing some related work in Sect. 2. In Sect. 3 a common definition and explanation of tokens, different kinds of tokens and their attributes follows. Section 4 introduces an implementation of a security framework and its actors. Afterwards, Sect. 5 discusses some major use cases of this implementation and how a faulty server configuration can be exploited. A solution to this problem is presented in Sect. 6 by illustrating a major use case that uses encrypted tokens. Next, Sect. 7 presents the introduced framework as a security layer in an agricultural environment. Finally, the paper concludes the results of this work in Sect. 8.

2 Related Work

In the publication [5] the authors analyze the *Oauth2.0* framework regarding possible security problems. Among others, they could identify several possible attacks which exploit an insufficient transport layer encryption (TLS). Unfortunately, the authors did not present a solution to this problem.

To reduce the harm of such attacks, [6] suggests to reduce the scope associated with an access token and to use a short expiration time for access tokens. However, the document does not provide an alternative to the TLS.

The *OpenID Connect* specification provides an identity authentication for OAuth2.0 systems. Although, OpenID Connect allows to use encrypted tokens for the user authentication [7], it does not define the communication for the authorization. As a consequence, the OAuth2.0 system that uses the OpenID Connect, may use plaintext tokens to authorize resource accesses.

In contrast to the OpenID Connect, our approach enforces encrypted token not only for the authentication of users but also for the authorization of resource accesses.

Similar Open-Source projects like *Apache Oltu* [8], *Restlet Framework* [9] and *APIs* [10] do not address the issue of weakly secured server communications.

An alternative approach that could be adopted for secure token communication is the blockchain technology. Blockchain technology is a decentralized approach. It was developed for transactions of electronic money within the *Bitcoin* network [11,12]. The downside of this technology is the increased implementation effort for new clients in the cloud. Further network traffic in this method would be higher, because every client additionally has to broadcast its token transactions to the so called minors.

3 Tokens

Tokens are often hardware-items for identifying and authenticating user. They also can be software-based artifacts of permission granting systems, where multipath authentication algorithms are used. In this work, the token concept of RFC 6750 [13] is used and implemented with the basic authentication concept of RFC 2617 [14]. Typically tokens are shared between multiple components, so every token has to be unique in its usage, autonomous and should not be repeated. Tokens themselves contain only implicit information – they carry information about the owner of the token (e.g. the user it was created for), the purpose of the token (e.g. for a user authentication), how long the token is valid etc.

Referring to the RFCs, the properties of different types of tokens are shown in Table 1:

perishable_token: is used to validate a single action; afterwards the token will be invalidated; it is secret and must be transferred via secure communication.

session_token: is valid for one specific session and can be used several times within this session; valid time span has to be renewed if token was used; session token must be transmitted over TLS.

access_token: can be used multiple times but cannot be renewed; its derivation must be verifiable; its transmission is encrypted.

refresh_token: can be used only once (must be invalidated after its use); like the `access_token` its derivation must be verifiable; the transmission of the token is encrypted.

Table 1. Token types

Type	Valid	Reuse	Renew	New on use
perishable_token	24 h	No	No	No
session_token	720 h	Yes	Yes	No
access_token	1 h	Yes	No	No
refresh_token	∞	No	No	Yes

4 Central Authentication & Authorization System (CAAS)

The Central Authentication & Authorization System (CAAS) is a security framework developed by the *FAW* (Institute for Application Oriented Knowledge Processing). The CAAS environment runs at least one authentication server and one authorization server. These servers handle access privileges for arbitrary users and clients to resources of one or more resource servers.

The authentication server checks the identity of users, whereas the authorization server manages privileges of users. A user uses a user-agent to access resources over a client or directly. Considering a web access, the web browser is a user-agent and the server providing the web page is a client accessing resources. The corresponding resource server asks the authorization server whether the user is authorized to access the resources. Because the authorization server cannot check the identity of the user, it asks the authentication server whether the user is who he claims to be. Therefore, the user has to prove his identity to the authentication server.

4.1 Authentication Server (ANS)

An Authentication Server (ANS) is responsible for the user management. Users can create accounts and reset their password. A user account can be in one of the following states: *activated*, *suspended*, *resumed* or *deleted*.

The main purpose of the ANS is to verify the identity of a user to the authorization server as described in the OpenID specification in [1]. Therefore, the server asks the user to prove his identity. Such identity prove can either be a password, a session token, or a perishable token (see Sect. 3). A session token is generated by the ANS and shared with a user that already authenticated himself with a password. This way the user does not have to send his password every time he has to authenticate.

Authenticated users can obtain an `authorization code`, which is basically a perishable token. Using this `authorization code` as described in the OAuth 2.0 standard, the user can provide information to the authorization server which only this user and the ANS know. The authorization server checks the identity of the user by letting the ANS validate the `authorization code`. Therefore, the user does not have to send his password or its session token to the authorization server to prove his identity.

4.2 Authorization Server (AZN)

The Authorization Server (AZN) administrates and provides information about which user has permission to access which resource. For this purpose it implements the OAuth2.0 protocol. Therefore, the server manages a so called `ScopeTree` whose nodes represent subjects and objects. In this context a subject can be a user, a client, or a group. An object is a resource or a group of resources provided by a resource server. A subject can have one or more permissions on an object.

The `ScopeTree` fulfills the following conventions:

- All clients must be listed in the `/clients` scope.
- Users must be listed in an `[ans_id]` scope.
- For every user in the system a link in the `/users` scope must exist.

Figure 1 shows an example of a `ScopeTree`. In the `/clients` scope is one client registered, `[client_id]`. For this client the AZN knows the public key and which scope nodes the client or its users are allowed to access. This example tree is paired with one ANS, `[ans_id]`. Therefore, the AZN knows the public key of `[ans_id]`. The AZN updates the user scope automatically when a request refers to a user that was not found in the scope tree. For this purpose, the AZN requests a list of all users from the registered ANS. Both users in the ANS scope have a link pointing to them in the `/users` scope.

Finally, the example lists some objects. Within the `[country_code]` scope is the object node `[object_id]` registered. This object has 2 sub objects, `[sub_object_id_1]` and `[sub_object_id_2]`. If a subject (e.g. `[user_id_1]`) has permissions for `object_id`, it has implicitly the same permissions on the sub objects. The inherited permissions on a sub object can be extended or removed. Because of this it is possible to create complex permission structures.

```
/
|-- /clients
|     |-- /[client_id]
|-- /users
|     |-- /[user_id_1] [LINK]
|     |     -> /^[ans_id]/[user_id_1]
|     |-- /[user_id_2] [LINK]
|     |     -> /^[ans_id]/[user_id_2]
|-- /[ans_id]
|     |-- /[user_id_1]
|     |-- /[user_id_2]
|-- /[country_code]/[object_id]
|     |-- /[sub_object_id_1]
|     |-- /[sub_object_id_2]
| ...
```

Fig. 1. Example `ScopeTree`

Permissions are stored as triples comprising of *subject*, *object* and a permission string. The *SCRUDL* scheme, used for the permission string, distinguishes the following six privileges:

- **(S)EARCH** in a requested scope and its subtree.
- **(C)REATE** a new object or resource in the requested scope.
- **(R)EAD** an existing object (including its attributes) in the requested scope.
- **(U)PDATE** or modify an existing object (including modification, deletion and creation of attributes) in the requested scope.
- **(D)ELETE** an existing object in the requested scope.
- **(L)IST** existing objects in a requested scope but not in its sub trees.

Figure 2 illustrates the format of the permission string. Every privilege has its fixed position in the string. If the letter representing the privilege is present, the privilege is granted. A `'.'` character instead of the letter means that the corresponding privilege of the parent scope is inherited. The `'-'` character denies the privilege it is representing, even if the privilege was inherited from the parent node. Example:

[..RU.-]: grants READ and UPDATE permission, SEARCH, CREATE, DELETE remain unchanged and the LIST permission is denied
[.C-.D.]: grants CREATE and DELETE permission, SEARCH, UPDATE, LIST remain unchanged and the READ permission is denied

```
permissions := ('S' | '.' | '-')('C' | '.' | '-')
               ('R' | '.' | '-')('U' | '.' | '-')
               ('D' | '.' | '-')('L' | '.' | '-')
```

Fig. 2. Permission format

The example in Fig. 3 illustrates a permission definition that belongs to the ScopeTree in Fig. 1. In the example the subject refers to the scope node /[ans_id]/[user_id_1] and the object refers to the scope node /[country_code]/[object_id]. As a consequence, the authorization server grants READ and UPDATE privileges for the user [user_id_1] to the object [object_id].

Since there is no permission definition for the child nodes, the user inherits there READ and UPDATE privileges for the child nodes.

```
Subject: /[ans_id]/[user_id_1]
Object: /[country_code]/[object_id]
Permissions: [..RU..]
```

Fig. 3. Permission example

5 Accessing Resources

This section explains some major work flows of the CAAS based on OAuth2.0 framework [2].

5.1 Obtaining Permissions

To obtain an authorization code, a User-Agent has to send an Authorization Request as illustrated in Fig. 4(a). Because the User-Agent is not authenticated, the AZN replies an Unauthorized Response (b). This response contains an URL to the ANS where the user should authenticate himself. After the user is successfully authenticated (c), the ANS responses with an Authentication Response that contains a perishable token (d). With this token the User-Agent can send an Approved Authorization Request (e). The AZN forwards the token to the ANS (f), which validates the token (afterwards the token is invalid for further requests) and replies an `Authentication Response` (g). At this point the AZN knows the identity of the user and sends back an `Authorization Code Response` if the user is allowed to access the resource (h). This `Authorization Code Response` contains a perishable token that is used to obtain an access token and a refresh token.

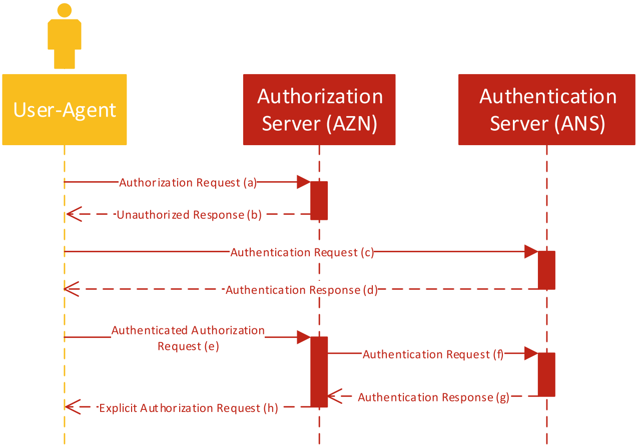

User-Agent
Authorization Server (AZN)
Authentication Server (ANS)
Authorization Request (a)
Unauthorized Response (b)
Authentication Request (c)
Authentication Response (d)
Authenticated Authorization Request (e)
Authentication Request (f)
Authentication Response (g)
Explicit Authorization Request (h)

Fig. 4. User authorization flow [2]

5.2 Permission Delegation

The permission delegation concept is based on the usage of bearer tokens as described in [13].

Figure 5 shows the process of obtaining and using access tokens and refresh tokens. The process starts with the `User-Agent` requesting a service from a `Client` (a). An `Authorization Challenge` responded by the `Client` contains a list of all scope permissions the `Client` needs to provide the service, as well as an URL to the responsible AZN (b). Using this `Authorization Challenge` the `User-Agent` obtains access to the required resources as demonstrated in Subsect. 5.1 (c–g).

The authorization code from the AZN is redirected as a Bearer token to the `Client` (h) and then forwarded back to the AZN (i). This way the user delegates his permissions on the resource to the `Client`. After validating the authorization code, the code is invalidated and the AZN replies with an access token and a refresh token (j). These tokens are used as bearer tokens to authorize resource access to the `Client` [13]. According to [2] the access token can be leaked, for example, by delegating it to a malicious `Client` (e.g. due to phishing). Therefore, the access token expires after a specified time to limit the harm of a stolen access token. Until the access token is expired, the `Client` can use it to access the resource (k, l). If the access token expires, the `Client` can request a new one from the AZN by providing the refresh token (m, n). The use of refresh tokens is described as optional in [2]. In case the refresh token is invalid, the `Client` proceeds with sending the `Authentication Challenge` to the `User-Agent` again (b).

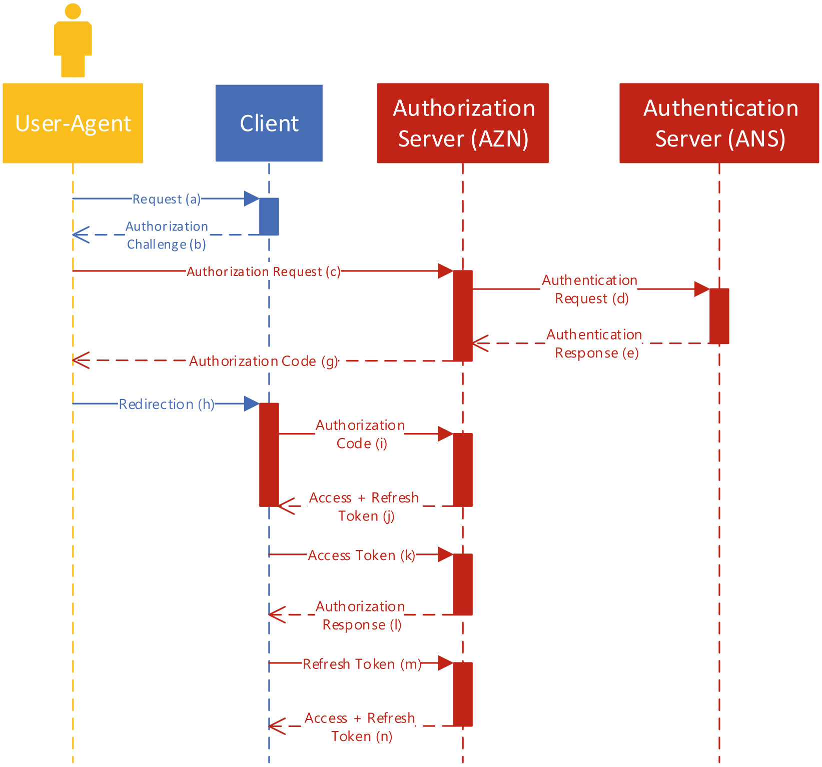

User-Agent
Client
Authorization Server (AZN)
Authentication Server (ANS)
Request (a)
Authorization Challenge (b)
Authorization Request (c)
Authentication Request (d)
Authentication Response (e)
Authorization Code (g)
Redirection (h)
Authorization Code (i)
Access + Refresh Token (j)
Access Token (k)
Authorization Response (l)
Refresh Token (m)
Access + Refresh Token (n)

Fig. 5. Access/refresh token sequence flow

This flow relies on secure communication between the single components. Although, the user can ensure that the communication between the `User-Agent` and other components is secured by the TLS protocol, an unsafe communication between `Clients`, AZNs and ANSs would be unrecognized. This would allow an attacker to steal the access token and get access to the requested resource.

5.3 Resource Access

This subsection extends the example in Subsect. 5.2 where the `Clients` require some resources from one or more `Resource Servers` to fulfill their task. Figure 6 illustrates this work flow. After the `User-Agent` sent the request (a), the `Client` tries to request the necessary resource (b). The `Client` does not need to try to request resources for which it has no authorization. Because the request

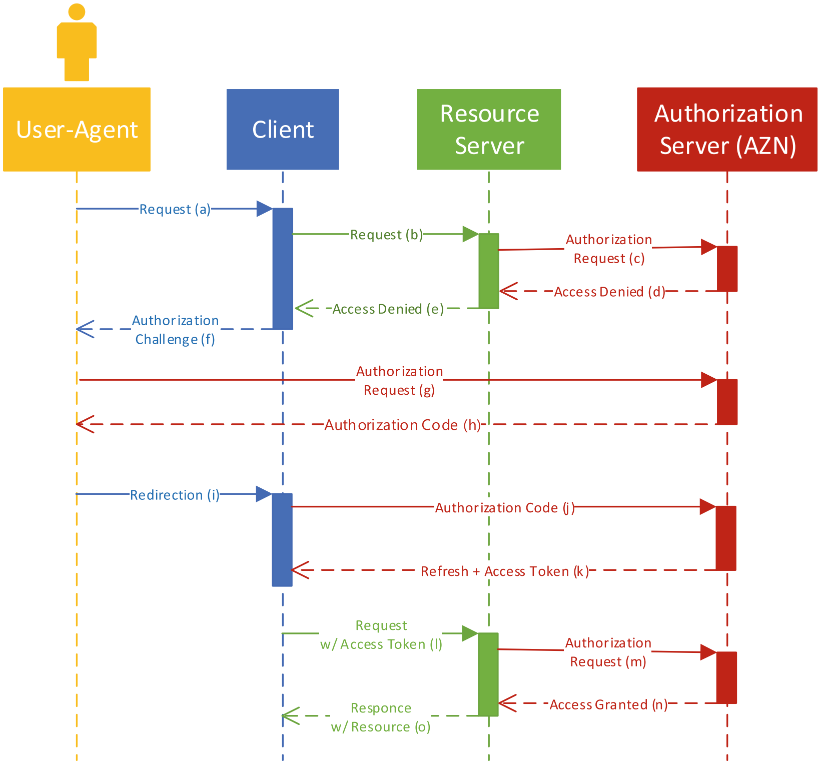


Fig. 6. Resource access sequence flow

(c) is not authorized, the AZN and further the Resource Server deny the access (d, e). Therefore, the Client returns an Authorization Challenge to the User-Agent (f). Like in Fig. 5, the User-Agent obtains an Authorization Code (g–i) which is used by the Client to get an access token from the AZN (i, k). Now, the Client can request the resource by use of the access token (l). The Resource Server uses the access token to obtain an Access Granted response from the AZN (m). After the Resource Server received the Access Grant (m), he returns the resource to the Client (o).

6 Secure Communication

In the previous sections, this paper presented the CAAS, its components and how they communicate with each other in token based manner. Further, a possible vulnerability of such implementations of OAuth2.0 and OpenID have been discussed. This section presents a solution to this problem.

By encrypting and/or signing the tokens before every transmission, the system is able to ensure secure communication between the single components independent from environmental influences like the server configuration.

To demonstrate this procedure, Fig. 7 illustrates the complete authorization flow from the initial user request to the final resource response.

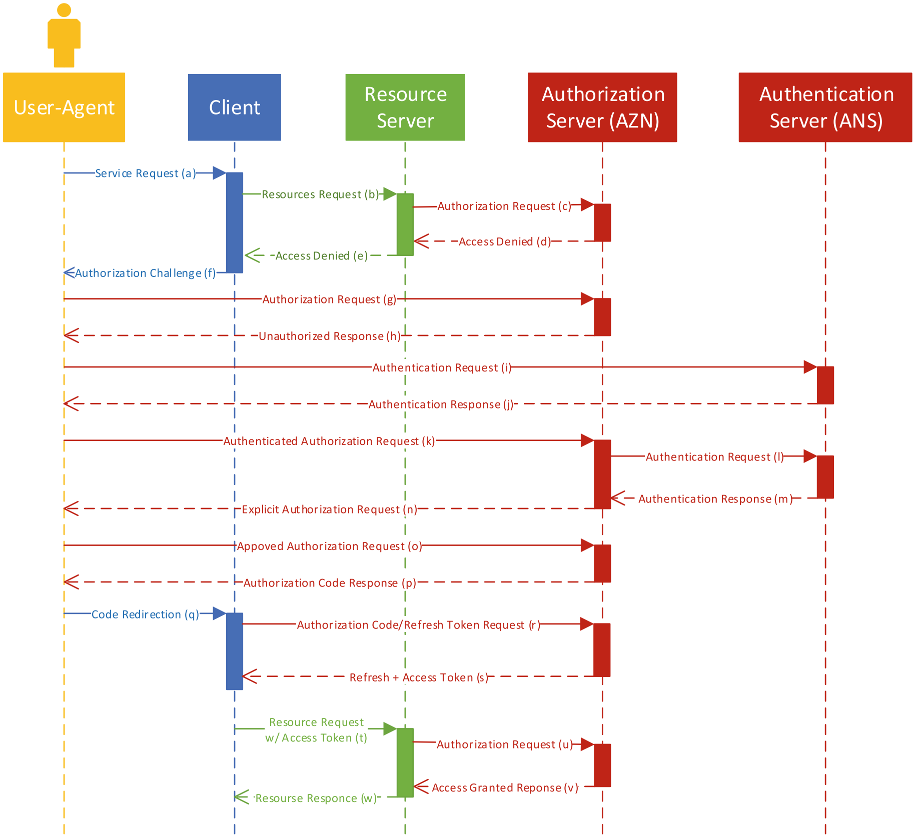

User-Agent
Client
Resource Server
Authorization Server (AZN)
Authentication Server (ANS)
Service Request (a)
Resources Request (b)
Authorization Request (c)
Access Denied (d)
Access Denied (e)
Authorization Challenge (f)
Authorization Request (g)
Unauthorized Response (h)
Authentication Request (i)
Authentication Response (j)
Authenticated Authorization Request (k)
Authentication Request (l)
Authentication Response (m)
Explicit Authorization Request (n)
Appoved Authorization Request (o)
Authorization Code Response (p)
Code Redirection (q)
Authorization Code/Refresh Token Request (r)
Refresh + Access Token (s)
Resource Request w/ Access Token (t)
Authorization Request (u)
Access Granted Reponse (v)
Resourse Responce (w)

Fig. 7. Complete delegation flow [2].

(a) The user requests a service provided by a `Client`. If the `Client` needs resources from one or more `Resource Servers` and has access tokens for those resources it will try to fetch those.

(b) The `Client` asks one or more `Resource Servers` for the resources it needs for its job. If the `Client` has no authorization information it will skip the steps (c) and (d).

(c) The `Resource Server` asks the `Authorization Server` if it can provide the requested resources to the `Client`.

(d) At this point, the `Authorization Server` denies the request (invalid/expired access token or no access token was provided).

(e) The `Resource Server` forwards the response to the `Client`.

(f) The Client issues an Authorization Challenge and returns it to the user. Such Authorization Challenge must contain the client_id and a list of requested scope nodes.

(g) The user forwards this to the Authorization Server. This can happen automatically using redirects [14].

(h) Since the user is not even authenticated yet, the Authorization Server replies with another Authorization Challenge but for an Authentication Server. It uses the client_id of the Authorization Server in this Authorization Challenge.

(i) The user forwards this to the Authentication Server, with his credentials. Those can be user_id/password, session token, etc. There will be at least 2 rounds between the Authentication Server and the user. First the user will get an Unauthorized response. In the next round, he provides his credentials. Before sending the credentials, the user can check if his browser is connected using the TLS protocol with an valid certificate. This way, the user ensures that he does not disclose his credentials.

(j) In case the second round will pass, the Authentication Server issues a specific access code for the Authorization Server and encrypts it with the Authorization Server's public key. Therefore, only the Authorization Server can use this token to validate the identity of the user.

(k) The user forwards this encrypted access code to the Authorization Server, which decrypts it, signs it and uses it to obtain an access token and a refresh token from the Authentication Server. These tokens are transmitted encrypted with the Authorization Server's public key as the access code before.

(l) The user authenticates himself to the Authorization Server directly using the token the Authentication Server issued to the user. The Authorization Server will then be able to check with the Authentication Server if the user is still valid and the authentication was not revoked by using the access token/refresh token issued to it by the Authentication Server. For the transmission of the access token/refresh token, these tokens must be encrypted by the Authentication Server's public key. Therefore, the Authentication Server an Authorization Server can be sure the tokens are not leak to a third party.

(m) The Authentication Server confirms or denies the request. For the confirmation, the server encrypts the already encrypted access token with its private key and returns it to the Authorization Server. This way, the Authorization Server can verify that the request confirmation was issued by the Authentication Server.

(n) The Authorization Server checks the original Authorization Challenge and renders a confirmation page to the user, where he reviews the permission delegation. It issues a perishable token with this response, which should be used to approve the request.

(o) If the user approves, he does so by sending the generated perishable token with the approved scope node back to the Authorization Server. Before sending the perishable token, the user can check if his browser is connected using the TLS protocol with an valid certificate. This way, the user ensures that he does not send the token to a third party in case of a Man-in-the-middle attack.

(p) If the perishable token was valid the Authorization Server issues an access code for the Client and encrypts it with the Client's public key. Therefore, only the Client is able request an access token using this code.

(q) The user forwards the encrypted access code to the Client.

(r) The Client decrypts the access code and uses it to obtain an access token and a refresh token directly from the Authorization Server. For this request, the access code must be encrypted with the Authorization Server's public key. Therefore, the Client an Authorization Server can be sure the token is not leak to a third party.

(s) The Authorization Server issues a new access token and refresh token, encrypts both with the Client's public key and sends them to the Client. Only the Client able to use these token to request a resource.

(t) The Client uses the access token encrypts with its Authorization Server's public key, every time it needs to access the given Resource Server. This way, the Authorization Server can verify that the access token was not leak to a third party.

(u) The Resource Server uses this signed access token to check with the Authorization Server if token is valid.

(v) The Authorization Server confirms or denies the request. For the confirmation, the server encrypts the already encrypted access token with its private key and returns it to the Resource Server. This way, the Resource Server can verify that the request confirmation was issued by the Authorization Server.

(w) The Resource Server makes the requested resources available to the Client.

In this work, the asymmetric Rivest-Shamir-Adleman (RSA) cryptosystem is used to prevent insecure token transmissions due to a faulty server configuration [15]. For the encryption we suggest keys with a minimum bit-length of 3072-bit. According to the NIST this key size is equivalent to a 128-bit symmetric key, which is acceptable for usage beyond 2030 [16].

7 Application in the Agricultural Domain

The CLAFIS (Crop, Livestock and Forests Integrated System for Intelligent Automation) project [17] is a research project, involving 12 partners from different countries and diversified fields of excellence, funded by the European Union. The project's goal is, to support people, who are working in the agricultural domain. On every step, the farmer should be assisted by modern technologies, via automation and analyzing data for predicting the agricultural returns. There are several sensors used, which provide real time data into the CLAFIS cloud for the knowledge processing system. With the results, e.g. regulation steps for machinery can be generated or predictions of the current disease pressure. Those results can be accessed by the farmer live via mobile devices.

There are many components, which have to communicate with each other, that a system like CLAFIS is able to work properly. In this context, security is a very important topic. On the one hand, valuable data must not be accessed by unauthorized parties, on the other hand, each component of the system must be able to trust every other component. The CAAS was designed to ensure these requirements in the CLAFIS.

For example: the Disease Pressure Model (DPM – calculation model for predicting the risk of the outbreak of a disease on a specific field), is working with data from several external systems and sensors. The implementation of the DPM is already secured by CAAS. Therefore, every component has to be authorized before it is able to access the DPM. Furthermore the components can trust on the certainty of the values provided by the DPM.

8 Conclusion

This paper presented the token-based authentication and authorization system, CAAS that implements the `OpenId` platform [1] and the `OAuth2.0` framework [2].

Afterwards, it showed the major use cases and discussed the importance of the Transport Layer Security (TLS) [18] for `OAuth2.0` systems. This leads to the result that a false configured server reveals a critical security flaw. During operation it is not possible for the user or the security framework to determine the missing security layer.

Therefore, a token encryption was introduced that secures the tokens and furthermore the resources of the system even if the TLS protocol is not activated. This mechanism prevents `OAuth2.0` based system from being exploited due to a deployment failure.

Finally, the use of the introduced CAAS and the token encryption within an agricultural domain was stated and the importance of secure authentication and authorization scheme was outlined.

Acknowledgement. This work was done as part of several projects by the authors during their stay at the Institute for Application Oriented Knowledge Processing at the Johannes Kepler University in Linz.

The research leading to these results has partly received funding from the European Union Seventh Framework Programme (FP7/2007–2013) under grant agreement no. 604659.

References

1. Recordon, D., Reed, D.: OpenID 2.0: a platform for user-centric identity management. In: Proceedings of the Second ACM Workshop on Digital Identity Management. DIM 2006, pp. 11–16. ACM, New York (2006)

2. Hardt, D.: The OAuth 2.0 Authorization Framework. RFC 6749, RFC Editor, October 2012
3. The Apache Software Foundation: SSL/TLS Configuration HOW-TO (2016). https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html#Introduction_to_SSL. Accessed 3 Sept 2016
4. Trustworthy Internet Movement: SSL Pulse - Survey of the SSL Implementation of the Most Popular Web Sites. https://www.trustworthyinternet.org/ssl-pulse. Accessed 3 Sept 2016
5. Yang, F., Manoharan, S.: A security analysis of the OAuth protocol. In: 2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), pp. 271–276, August 2013
6. Lodderstedt, T., McGloin, M., Hunt, P.: OAuth 2.0 Threat Model and Security Considerations. RFC 6819, RFC Editor, January 2013
7. Sakimura, N., Bradley, J., Jones, M.B., de Medeiros, B., Mortimore, C.: OpenID Connect Core 1.0. The OpenID Foundation, S3 (2014)
8. The Apache Software Foundation: Apache Oltu: An OAuth Open Source framework. https://cwiki.apache.org/confluence/display/OLTU/Index (2013). Accessed 3 Sept 2016
9. RestLet Inc.: RestLet Framework (2016). https://restlet.com/technical-resources/restlet-framework/guide/2.3/extensions/oauth. Accessed 3 Sept 2016
10. Harsta, O.: OAuth-Apis: OAuth Authorization as a Service (2012–2016). https://github.com/OAuth-Apis/apis. Accessed 3 Sept 2016
11. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008). http://bitcoin.org/bitcoin.pdf. Accessed 3 Sept 2016
12. Travis, P.: The Bitcoin Revolution: An Internet of Money. Travis Patron (2015) Accessed 3 Sept 2016
13. Jones, M.B., Hardt, D.: The OAuth 2.0 Authorization Framework: Bearer Token Usage. RFC 6750, RFC Editor, October 2012
14. Franks, J., Hallam-Baker, P.M., Hostetler, J.L., Lawrence, S.D., Leach, P.J., Luotonen, A., Stewart, L.C.: HTTP Authentication: Basic and Digest Access Authentication. RFC 2617, RFC Editor, June 1999
15. RSA Security: Information Security, Governance, Risk, and Compliance - EMC (2014). http://www.rsa.com. Accessed 3 Sept 2016
16. Barker, E., Barker, W., Burr, W., Polk, T., Smid, M., Zieglar, L.: NIST Special Publication 800-57 Revision 4 Recommendation for Key Management Part 1: General (2016). http://dx.doi.org/10.6028/NIST.Spp.800-57pt1r4
17. CLAFIS Project: CLAFIS: crop, livestock and forests integrated system for intelligent automation (2013–2016). http://www.clafis-project.eu EU Seventh Framework Programme NMP.2013.3.0-2
18. Dierks, T., Rescorla, E.: The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246, RFC Editor, August 2008

An Enhancement of the Rew-XAC Model for Workflow Data Access Control in Healthcare

Thanh Tien Nguyen[1](✉), Nguyen Hoang Nam Pham[2],
and Que Nguyet Tran Thi[3]

[1] IT and Data Management Department,
Oxford University Clinical Research Unit, Hà Nội, Vietnam
tiennt@oucru.org
[2] Department of Information Technology,
University of Economics, Ho Chi Minh City, Vietnam
nam@ueh.edu.vn
[3] Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh City, Vietnam
ttqnguyet@cse.hcmut.edu.vn

Abstract. The Rew-XAC model, based on Extensible Access Control Markup Language (XACML) 3.0, has been developed to solve the problem in the case that requests receive *"Not Applicable"* responses from the policy decision point (PDP). According to the most applicable policy that has the best score computed by a fuzzy function, the Rew-XAC model carried out rewriting the request. However, an important issue not addressed yet in the Rew-XAC model is that there has more than one policy with the same highest fuzzy value. In this paper, we propose an enhancement that assigns a union operator for all resource filter expressions produced from the related modules in the Rew-XAC model for each selected policy to the rewritten request. Besides, we demonstrate the potential of our model through analyzing the complex security requirements for a case study in the healthcare domain, and then propose a mechanism integrated with the proposed model to support access control for workflow data. We also perform an experiment using the dataset of policies in the case study to verify the feasibility of our approach in the healthcare domain that needs the data-protection rigorously complying with the regulations.

Keywords: Access control · Rew-XAC · Rewriting · Workflow data · XACML

1 Introduction

Protecting resources from unauthorized accesses is an aspect on which most of access control models concentrate. An access control policy defines conditions to determine which access resources can be granted and to whom. *Attribute based Access Control (ABAC)* model is defined as *an access control method where subject requests to perform operations on objects are granted or denied based on assigned attributes of*

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 251–263, 2016.
DOI: 10.1007/978-3-319-48057-2_18

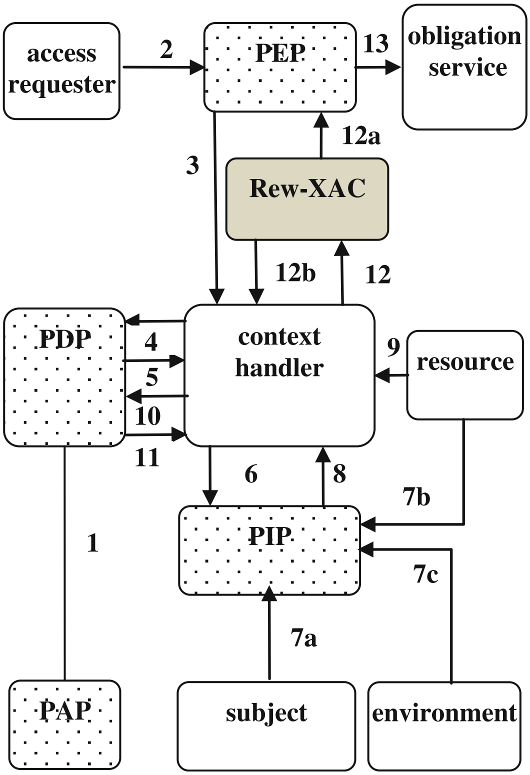
the subject, assigned attributes of the object, environmental conditions, and a set of policies that are specified in terms of those attributes and conditions [10]. In addition, Organization for the Advancement of Structured Information Standards (OASIS) has developed the eXtensible Access Control Markup Language (XACML) standard [6] supporting specifying attribute based policies and requests based on attributes of subjects, actions, environments and resources. In XACML, each specific request receives one of four results from the access decision-making process, including Permit, Deny, Indeterminate and Not Applicable [5]. The paper [1] concentrated on analyzing the request in the case of "Not Applicable" response and introduced the Rew-XAC model, which modifies such kind of request. Based on conditions of system's policies, the system allows the request to access data with some limitations in comparison with the original one. We realize that this model needs an enhancement to enable a high efficient, flexible and applicable mechanism for data access control in healthcare.

Data access is an essential requirement to biomedical research, clinical education, and patient care [3]. In healthcare information systems, the enormous amount of medical records has additional value when they are stored in data warehouses [9]. These records contain a great deal of data about people and may also contain privacy information such as fertility and abortions, emotional problems and psychiatric care, sexually transmitted diseases, HIV status, substance abuse, physical abuse, genetic predisposition to diseases, and so on [5]. Using such information must strictly follow Health Insurance Portability and Accountability Act (HIPAA) regulations [2] that state many rigorous data-protection policies. Hence, a repository of policies for accessing resources in these data warehouses needs to be introduced to prevent unauthorized access. However, these systems must provide data access control mechanisms that allow right people to access right information in the right context so that healthcare providers would deliver proper services to patients without misusing sensitive information. The workflow system for the clinical trial study [12] also needs access the data warehouse and states such complex security requirements, which this paper will analyze.

The rest of paper is organized as follows: Section 2 analyzes the Rew-XAC model, which introduces the request rewriting approach based on the XACML 3.0 model. Section 3 shows the limitations of the Rew-XAC model, and then describes an enhancement to improve the Rew-XAC model. Section 4 introduces a case study in healthcare domain, analyzes the regulations to show potential of enhanced Rew-XAC model integration. Section 5 presents an experiment of access control for workflow data in a case study to demonstrate and verify the proposed enhancement. Section 6 consists of conclusions and recommended future works.

2 Related Work

In the XACML standard, ABAC policies and requests are specified based on attributes of subjects, actions, environments and resources. In some certain situations, the requester receives a "*Not Applicable*" decision when the system cannot find any appropriate policy for evaluating the request. The numerous of "*Not Applicable*" response may reduce the availability of system. To tackle this problem, the Rew-XAC model [1] modifies the original XACML model by adding a new interceptor module



ID	*Description*
1	Policy
2	Access request
3	Request
4	Request notification
5	Attribute queries
6	Attributes query
7a	Subject attributes
7b	Resource attributes
7c	Environment attributes
8	Attribute
9	Resource content
10	Attributes
11	Response context
12	Response
12a	I) Permit; Deny or Indeterminate response II) Original response from 12
12b	Rewritten request
13	Obligations

Fig. 1. The Rew-XAC Model

shown in Fig. 1 for rewriting requests with “Not Applicable” response based on conditions specified in policies. This model supports for the system to allow the request with some limitations in comparison with the original one.

In the model, the Rew-XAC module receives the input request having “Not Applicable” response. It chooses a suitable policy having the highest fuzzy value to replace the resource filter expressions on the set-theoretic intersection between the Resource of that policy and the one of the request to make a rewritten request. This module then forwards the rewritten request to Context Handler like a normal flow. The system will terminate the loop when there is no suitable policy found to rewrite the request or the decision value of response that is not “Not Applicable”. Even when the system policies are changed before receiving a request or after having one created rewritten request will be used in the rewritten progress, so that assures the dynamic policies enforcement.

The following formula describes how the fuzzy value for a policy calculated:

$$Fuzzy_{Policy} = Fuzzy_{Sub} * Fuzzy_{Act} * Fuzzy_{Env} * Fuzzy_{Res}$$

Group A {subject, action, and environment} the components of Policy is subset group B {subject, action, and environment} the components of Request:

$$If(A \subseteq B) \Leftrightarrow Fuzzy_A = 1 \; Else \; Fuzzy_A = 0$$

$$Fuzzy_{Res} = |Res_{Policy} \cap Res_{Request}| * \frac{1}{n}$$

Where n is the number of component of Resource Request.

In this approach, a suitable policy is the one having conditions on attributes in categories Subject, Action and Environment satisfied all by the request, and has the highest fuzzy value computed on set-theoretic intersection between Policy Resource and Request Resource.

3 An Enhancement for the Rew-XAC Model

In theory, Rew-XAC may ignore other suitable policies having the same computed fuzzy value against the chosen one. Table 1 shows an example for this issue.

In Table 1, *p1* and *p3* are two policies having the same fuzzy value *0.5* calculated for the request *r*. If the system uses *p1* for rewriting the request, the next process of rewriting request will ignore the policy *p3* because of HIV status in *p3* is definitely different to the one in *p1*. Then, the requester may never able to reach the resource

Table 1. An example of a request having more than one suitable policy in the Rew-XAC

	Subject	Action	Environment	Resource
p1	Doctor	Read	Time: 08:00 AM–16:00 PM	*ResourceName:* Hematology test results *Filter Expressions:* HIV status: negative Treatment department: Infection control
p2	Doctor nurse	Read	Time: 08:00 AM–16:00 PM	*ResourceName:* Biochemistry test results *Filter Expressions:* Treatment department: Intensive Care Unit
p3	Doctor	Read	Time: 08:00 AM–16:00 PM	*ResourceName:* Hematology test results *Filter Expressions:* HIV status: positive Patient age: >12
r	Doctor	Read	Time: 09:00 AM	*ResourceName:* Hematology test results *Filter Expressions:* Patient age: >12 Dengue confirmed: Yes Treatment department: Infection control
r_1	Doctor	Read	Time: 09:00 AM	{*ResourceName:* Hematology test results}, {*ResourceName:* Biochemistry test results}

defined in *p3* if the same request statement is used. Therefore, this case causes a problem in implementing the Rew-XAC model.

Besides, Rew-XAC also does not support multiple decisions whereas the resource component of request may be an array of filter expressions in practical such as *r1* in Table 1. In this section, we propose an enhancement for Rew-XAC to tackle to these problems.

In the XACML architecture [5], PEP sends an authorization request to PDP using a XACML context, which contains the subject, action, resource, and environment. We propose using an array of filter expressions for the resource component instead of a single-valued in order to facilitate the creation of multiple decisions. Besides, our approach is retrieving the suitable system policies for each filter expressions in the resource component of the request that receives "Not Applicable" decision to support rewriting request.

Let $R = \{rq_1, rq_2, \ldots, rq_n\}$ is the set of resource filter expressions for request *r*.

- n is the number of resource filter expressions for request
- rq_i $(1 \leq i \leq n)$ is a specific expression belonging to *R*.

$P_i = \{p_{i1}, p_{i2}, \ldots, p_{im}\}$ is the set of policies having the best fuzzy value computed by the fuzzy function of Rew-XAC, and suitable for rq_i.

- m is the number of suitable polices for rq_i
- p_{ij} $(1 \leq j \leq m)$ is a specific policy belonging to P_i.
- The $Fuzzy_{Policy}$ value of policies must be greater than 0.

Res_{ij} is the specific filter expression built from replacing the resource of a condition on the set-theoretic intersection between the Resource of policy p_{ij} and the specific filter expression rq_i of the request *r*.

Let assume *R'* is the set of resource filter expression of the rewritten request *r'*. If there is any decision in the response having "Not Applicable" value, the Enhanced Rew-XAC module will do following steps:

Step 1. For each specific expression rq_i in R, the module finds P_i the set of policies having the best fuzzy value computed by the fuzzy function of Rew-XAC, and suitable for rq_i.
Step 2. For each policy p_{ij} in P_i, the module builds Res_{ij} and adds Res_{ij} into *R'*.
Step 3. The module then forwards the rewritten request *r'* back to the context handler.

The loop will stop if there is no suitable policies exist or the response does not contain any "Not Applicable" decision.

Example 1: Let consider using the fuzzy function of Rew-XAC for two policies in the Table 1 with the request *r*. The computed fuzzy value for each policy will be *p1 = p3 = 0.5*. If the original Rew-XAC module chooses *p1* for making the rewritten request *r*', the policy *p3* may be ignored in the next turn unless *p3* changed to *p3'* before the next turn happened and *p3'* has the highest computed value. Therefore, the requester could only receive the granted access for the resource described in *p1* and may not be able to reach the resource described in *p3*. If the enhanced model is applied, the resource requirement is an array instead of single-valued in order to facilitate the creation of multiple decisions. Therefore, it allows the requester able to reach a part of resource described in both *p1*and *p3.* The rewritten request *r'* is described in Table 2.

In fact, the enhanced model works similar to the original one if there is only one suitable policy for the request. For the certain circumstance, the enhanced model allows the requester reach resources which the suitable policies indicate, whereas the original one only allows the requester accessing the resource defined in the first suitable policy. Therefore, the enhanced one is more flexible, still ensuring the system regulations, and inheriting the dynamic policies support of the original one.

Table 2. An example of a request, policies and rewritten one of Enhanced Rew-XAC

	Subject	Action	Environment	Resource
p1	Doctor	Read	Time: 08:00 AM–16:00 PM	*ResourceName*: Hematology test results *Filter Expressions:* HIV status: negative Treatment department: Infection control
p3	Doctor	Read	Time: 08:00 AM–16:00 PM	*ResourceName*: Hematology test results *Filter Expressions:* HIV status: positive Patient age: >15
r	Doctor	Read	Time: 09:00 AM	*ResourceName*: Hematology test results *Filter Expressions:* Patient age: >12 Dengue confirmed: Yes Treatment department: Infection control
r'	Doctor	Read	Time: 09:00 AM	{*ResourceName*: Hematology test results *Filter Expressions:* Patient age: >12 Dengue confirmed: Yes Treatment department: Infection control HIV status: negative}, {*ResourceName*: Hematology test results *Filter Expressions:* Patient age: >15 Dengue confirmed: Yes Treatment department: Infection control HIV status: positive}

4 A Case Study

In this section, we discuss some requirements relating to data management in a clinical trial study described in the article [12].

The study is conducted at four research sites in Vietnam, including Ho Chi Minh City, Hanoi, Hai Phong, and Quang Ninh. Each member involving this study has a role and assigned into a research site. There are six roles in this study, including patient, principle investigator, study doctor, pharmacist, study coordinator, research nurse and data manager. The Fig. 2 shows the tasks and relationship in workflow process of the study. A research nurse or a study doctor interviews and consults the patient who benefits in the study. The patient must complete the consent form when participates the study. After screening results are available, eligibility for this treatment protocol will be assessed by the study inclusion and exclusion criteria. Then, a nurse or a doctor interviews patient for some basic information, including demographic information and allergy information for the baseline stage. Following that, the nurse completes the physical examination for patient at day 1, including height, weight, and blood pressure and pulse. If there is any allergy information reported, a study pharmacist will involve into the medicine-consulting step. After having the report from pharmacist, the study doctor takes the treatment randomization for the patient, orders laboratory tests for patient and follows up the patient evolutions. If there is any adverse event during the treatment process, a study doctor produces an adverse event report and delivers it to the principle investigators. The clinical response will also be evaluated earlier during therapy at week 2, week 4, and later at week 24. Finally, the outcome report is produced for that patient.

This study uses a clinical data management system that supports workflow implementation using a XACML policy repository named Workflow Policy Repository (WFPR) for workflow access control. In this system, an electronic Case Report Form (eCRF) is designed to collect the patient data for a domain in a clinical trial; and it denotes a tuple with object attributes. All eCRFs produced from the system is stored in

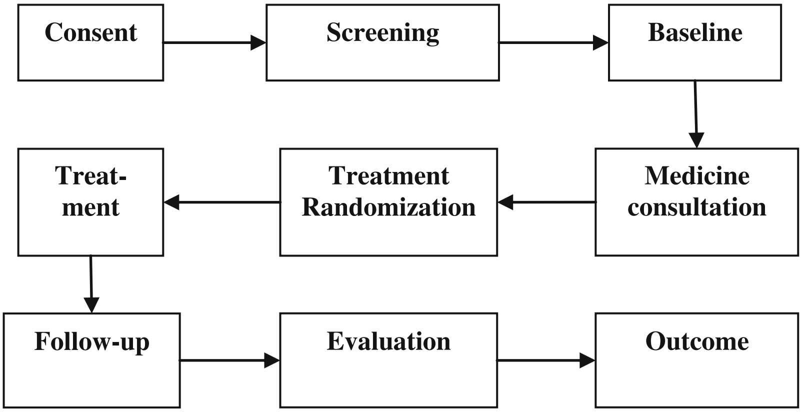


Fig. 2. Study Flow Diagram

a data warehouse, which has a policy repository named Data Warehouse Policy Repository (DWPR) for data access control.

Assuming the current system manages access control by using two above policy repositories. We discuss some typical regulations that the study protocol may stipulate and means by which the workflow system could support them effectively.

4.1 Regulations in Case Study

There are dozens of complex security requirements in the case study for workflow data access control. In this paper, we only discuss some typical requirements which ask the access control model for rewriting request module integration to be satisfied.

Regulation 1. *When perform the Treatment, Follow-up or Evaluation task, a doctor d can read the Case Report Forms (CRFs) of*

- *the patient p that d is treating (may include those records of p that were not treated by d), and*
- *those cases of other patients besides p the doctor d treated before.*

Discussion: In this requirement, at least two access control rules may effect on the requests. The first one is a policy wfa-p1 in WFPR requiring the attribute "task" for the environment component. Besides, a policy wfda-p1 in DWPR does not require this attribute because the other systems, which may not support workflows but require access data in the data warehouse. However, wfda-p1 may require the other different attributes that may not be in the resource component of the request when performing workflow. Table 3 shows the typical example of case study.

In Table 3, the policy wfda-p1 for data warehouse defines that a doctor can read all medical records of a patient who has HIV test and agrees to this, whereas the request r from tasks of workflow usually missing the "Patient Agreed" attribute because the doctor already has the patient consent form. Therefore, wfda-p1 may cause preventing doctor d access the CRFs of patient p when d performs these tasks in the workflow.

Due to this circumstance, the Enhanced Rew-XAC model can effectively support solving this restriction. It uses these both policies to evaluate the request and makes a new request satisfying these policies. The example in the Table 3 shows the rewritten request *r'* reduced required resource for aligning with policies wfda-p1 and wpa-p1.

Regulation 2. *A pharmacist ph can access allergy information, examination, adverse event and serious adverse event CRFs of the patient p when performing the medicine consultation task. This pharmacist also needs other related information about the cases he/she consulted for an effective consultation.*

Discussion: This requirement is similar to the requirement in Regulation 1. The pharmacist needs to retrieve the historical record of patients related to the medicine consulting session. The protecting mechanism of access control for data warehouse causes the major difficulty for this requirement because there is a policy at Data Warehouse Polices Repository, which may not contain any information about the task.

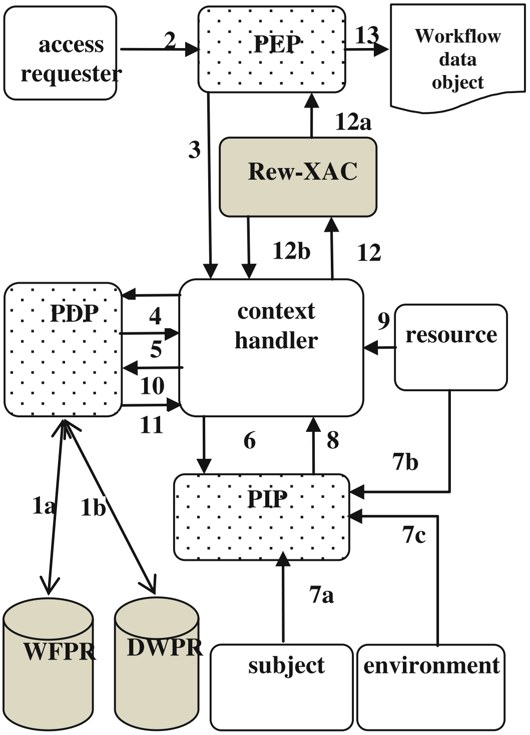
Table 3. An example of a request and a rewritten one of Rew-XAC

	Subject	Action	Environment	Resource
wfa-p1	Doctor	Read	Time: 08:00 AM–16:00 PM Task: Treatment	*ResourceName*: Medical Records *Filter Expressions:* HIV test performed: Yes Consent: Yes
wfda-p1	Doctor	Read	Time: 08:00 AM–16:00 PM Network: Internal	*ResourceName*: Medical Records *Filter Expressions:* HIV test performed: Yes Patient agreed: Yes Treatment department: Infection control
r	Doctor	Read	Time:09:00 AM Task: Treatment Network: Internal	*ResourceName*: Medical Records *Filter Expressions:* Treatment department: Infection control
r’	Doctor	Read	Time:09:00 AM Task: Treatment Network: Internal	{*ResourceName*: Medical Records *Filter Expressions:* Treatment department: Infection control HIV test performed: Yes Consent: Yes}, {*ResourceName*: Medical Records *Filter Expressions:* Treatment department: Infection control HIV test performed: Yes Patient agreed: Yes}

This requirement can be satisfied if the Enhanced Rew-XAC model is integrated, and the system uses policies in both repositories, including DWPR and WFPR to evaluate the request. When the decision of response is the “Not Applicable” value for the request, the rewriting request mechanism enables and uses policies in both repositories to reduce required resource for aligning with them.

4.2 An Enhanced Rew-XAC

The requirements in Regulations 1 and 2 show the necessary of the Enhanced Rew-XAC model for access control of workflow data. We analyze the requirement of



ID	*Description*
1a	Workflow policy
1b	Data warehouse policy
2	Access request
3	Request
4	Request notification
5	Attribute queries
6	Attributes query
7a	Subject attributes
7b	Resource attributes
7c	Environment attributes
8	Attribute
9	Resource content
10	Attributes
11	Response context
12	Response
12a	I) Permit; Deny or Indeterminate response II) Original response from 12
12b	Rewritten request
13	Workflow data object

Fig. 3. Proposed architecture for workflow data access control with the Enhanced Rew-XAC

case study and propose architecture for workflow data access control integrating the Rew-XAC module to support rewriting the request in Fig. 3.

In the proposed architecture, the workflow system with a dedicated data warehouse relies on two policy repositories to make decisions for data access control, including Data Warehouse Policy Repository (DWPR) and Workflow Policy Repository (WFPR). The DWPR is a repository of policies that supports data access control at the data warehouse. The systems accessing the data warehouse must strictly follow these policies despite the fact that the workflow system for case study is accessing its data at this warehouse. Besides, the WFPR is a repository of policies of the workflow system, which is defined to indicate the conditions and the resources needed to perform the tasks. For each the data access request from this workflow system, the access control module evaluates the request and finds the suitable policies in both policy repositories to make the response for the request. In this situation, the numerous of policies may cause the issues we present in Sect. 3. Besides, the requester must provide all information for attributes in each filter expression to satisfy the policies having many filter expressions in the resource component to avoid the "Not Applicable" decision. For this certain circumstance, the Enhanced Rew-XAC module in this architecture can support rewriting the request aiming to reduce the times of "Not Applicable" response receiving, allowing requesters spend less effort to retrieve data, but ensuring policy satisfaction.

For the non-workflow request, only data warehouse policies are applicable used to evaluate the request in theory. However, the system must check the policies from WFPR if the data is produced from a workflow instance to detect any changes of policies, so that makes a niche response for the request.

5 Experiment

To verify the applicability of the enhanced model, we gathered the policies from the system used for case study in Sect. 4, including 50 policies for workflow access control and 80 policies for workflow data. To support the evaluation, we built six sets of request showed in Table 4 to access resources in data warehouse.

In the first scenario, we performed finding out the numbers of “Not Applicable” response using AT&T XACML module for each set of requests. Then, we integrated the Enhanced Rew-XAC module into AT&T XACML to find the numbers of “Not Applicable” response to compare to the original ones. Through this comparison, we could see how the enhanced new model done. In the next scenario, we counted the times that the system detected policies having the same best fuzzy value when the rewriting process performed. This result in this experiment showed the necessary of our enhancement for the Rew-XAC model. Table 4 shows the results of experiment.

In this experiment, we could not measure how the original Rew-XAC done for these policy and request datasets because this model did not support multiple decisions for the response and could not stop the loop. The paper [1] clearly demonstrates that Rew-XAC can support dynamic policy changes, and the enhanced model is only a one extending the Rew-XAC. Therefore, we did not measure how the enhanced model supports dynamic policy changes in this experiment.

Table 4. Sets of request and results of the measurement

Number of request	Number of not applicable response without enhanced Rew-XAC	Number of not applicable response after integrating enhanced Rew-XAC	Request rewriting time with enhanced Rew-XAC	The number of fuzzy value duplication detected in enhanced model
500	464	243	225	13
1000	949	547	419	20
2000	1876	1093	825	51
5000	4705	2577	2224	155
10000	9420	5161	4424	269
20000	18787	10434	8698	577

6 Conclusion and Future Work

On the assumption that other unpredictable factors do not adversely affect the result in our experiment, we conclude that the enhanced model effectively reduces a numerous "Not Applicable" responses for the case study. Besides, the experiment results clearly stated that the Rew-XAC model had an issue of fuzzy value duplication addressed in Sect. 2. We have proposed an enhancement, so that tackles the problem of Rew-XAC model and allows the enhanced one inherit the powerful feature of the original one such as dynamic policies support. This paper also provided the workflow data access control model in healthcare domain which was a comprehensive architecture and highly applicable. Unfortunately, the experiment showed a numerous rewriting request times, which affected the system performance. However, our proposed model could satisfy the complex security requirements of stakeholders to access the workflow data in the data warehouse, but ensuring the data-protection regulations.

An extension to this work is to develop a method to support the policy changes before sending the response for rewritten request to ensure the ABAC policy enforcement in dynamic environments. We also plan to analyze all risks related to workflow data misusing in the proposed architecture and develop a methodology to detect and reduce it.

Acknowledgments. This research is funded by Ho Chi Minh City University of Technology, Vietnam National University - Ho Chi Minh City (VNUHCM) under grant number T-KHMT-2015-29.

References

1. Ha, X.S., Tran, L.K., Dang, T.K., Küng, J.: Rew–XAC: an approach to request rewriting for elastic ABAC enforcement with dynamic policies (2016, in submission)
2. Health Insurance Portability and Accountability Act: (n.d.). https://en.wikipedia.org/wiki/Health_Insurance_Portability_and_Accountability_Act. Accessed 29 Apr 2016)
3. Zhang, Y., Dhileepan, S., Schmidt, M., Zhong, S.: Emergency access for online personally controlled health records system. Inform. Health Soc. Care **37**(3), 190–202 (2012). doi:10.3109/17538157.2011.647935
4. Elkandoussi, A., Elbakkali, H.: On access control requirements for inter-organizational workflow. In: Proceedings of the 4th Edition of National Security Days (JNS4) (2014). doi:10.1109/jns4.2014.6850128
5. Shengli, W., Amit, S., John, M., Zongwei, L.: Authorization and access control of application data in workflow systems. J. Intell. Inf. Syst. **18**, 71–94 (2002). doi:10.1023/A:1012972608697
6. XACML (n.d.). https://en.wikipedia.org/wiki/XACML. Accessed 29 Apr 2016
7. Barker, S., Fernández, M.: Term rewriting for access control. In: Damiani, E., Liu, P. (eds.) Data and Applications Security XX. LNCS, vol. 4127, pp. 179–193. Springer, Heidelberg (2006). doi:10.1007/11805588_13
8. Thi, Q.N., Dang, T.K.: X-STROWL: a generalized extension of XACML for context-aware spatio-temporal RBAC model with OWL. In: Seventh International Conference on Digital Information Management (ICDIM 2012) (2012). doi:10.1109/icdim.2012.6360113

9. Thadani, S.R., Weng, C., Bigger, J.T., Ennever, J.F., Wajngurt, D.: Electronic screening improves efficiency in clinical trial recruitment. J. Am. Med. Inform. Assoc. **16**(6), 869–873 (2009). doi:10.1197/jamia.m3119
10. Hu, V.C., Ferraiolo, D., Kuhn, R., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K.: Guide to Attribute Based Access Control (ABAC) Definition and Considerations (2014). doi:10.6028/nist.sp.800-162
11. Anderson, A.H.: A comparison of two privacy policy languages. In: Proceedings of the 3rd ACM Workshop on Secure Web Services - SWS 2006 (2006). doi:10.1145/1180367.1180378
12. Le, T. (n.d.): Itraconazole Versus Amphotericin B for the treatment of Penicilliosis (IVAP). http://isrctn.org/

Access Control in NoSQL and Big Data

ASASPXL: New Clother for Analysing ARBAC Policies

Anh Truong[1](✉) and Silvio Ranise[2]

[1] Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
anhtt@hcmut.edu.vn

[2] Security and Trust Unit, FBK-Irst, Trento, Italy
ranise@fbk.eu

Abstract. Access Control is becoming increasingly important for today's ubiquitous systems. In access control models, the administration of access control policies is an important task that raises a crucial analysis problem: if a set of administrators can give a user an unauthorized access permission. In this paper, we consider the analysis problem in the context of the Administrative Role-Based Access Control (ARBAC), one of the most widespread administrative models. We describe how we design heuristics to enable an analysis tool, called ASASPXL, to scale up to handle large and complex ARBAC policies. An extensive experimentation shows that the proposed heuristics play a key role in the success of the analysis tool over the state-of-the-art analysis tools.

Keywords: User-role reachability problem · Administration · Safety analysis · Access control · Model checking · Heuristics · Security

1 Introduction

Modern information systems contain sensitive information and resources that need to be protected against unauthorized users who want to steal it. The most important mechanism to prevent this is Access Control [7] which is thus becoming increasingly important for today's ubiquitous systems. In general, access control policies protect the resources of the systems by controlling who has permission to access what objects/resources.

Role-Based Access Control (RBAC) [14] is one of the most widely adopted access control models in the real world. In RBAC, access control policies specify which users can be assigned to roles which, in turn, are granted permissions to perform certain operations in the system. Usually, RBAC policies need to be evolved according to the rapidly changing environments and thus, it is demanded to have some mechanisms to control the modification of the policies. Administrative RBAC [6] is the corresponding widely used administrative model for RBAC policies. The main idea of ARBAC is to provide certain specific users, called administrators, some permissions to execute operations, called administrative

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 267–284, 2016.
DOI: 10.1007/978-3-319-48057-2_19

actions, to modify the RBAC policies. In fact, permissions to perform administrative actions must be restricted since administrators can only be partially trusted. For instances, some of them may collude to, inadvertently or maliciously, modify the policies (by sequences of administrative actions) so that untrusted users can get sensitive permissions. Thus, automated analysis techniques taking into consideration the effect of all possible sequences of administrative actions to identify the safety issues, i.e. administrative actions generating policies by which a user can acquire permissions that may compromise some security goals, are needed.

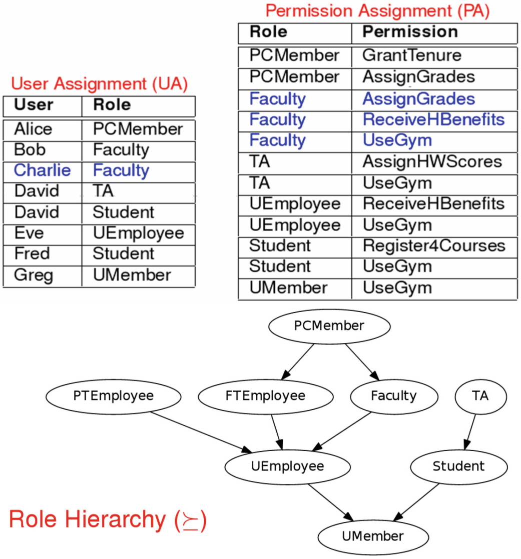
Several automated analysis techniques (see, e.g., [4,8,11,12,16,17]) have been developed for solving the user-role reachability problem, an instance of the safety issues, in the ARBAC model. Recently, a tool called ASASPXL [13] has been shown to perform better than the state-of-the-art tools on sets of benchmark problems in [10,16]. The main advantage of the analysis technique inside ASASPXL over the state-of-the-art techniques is that the tool can solve the user-role reachability problem with respect to a finite but unknown number of users in the policies manipulated by the administrative actions. However, ASASPXL does not scale to solve problems in some recently proposed benchmarks in [17]. This is because the so-called state explosion problem has not been handled carefully and thus, prevent ASASPXL to tackle such benchmarks.

In this paper, we study how to design heuristics to enable ASASPXL to analyze large and complex instances of user-role reachability problems. The main idea is to try to alleviate the state explosion problem, which is well-known problem in model checking techniques, in the analysis of ARBAC policies. We also perform an exhaustive experiment to conduct the effectiveness of proposed heuristics and compare ASASPXL's performance with the state-of-the-art analysis tools.

The paper is organized as follows. Section 2 introduces the RBAC, ARBAC models, and the related analysis problem. Section 3 briefly introduces the automated analysis tool ASASPXL and the model checking technique underlying it. The proposed heuristics to enable ASASPXL to scale to solve user-role reachability problem are described in Sect. 4. Section 5 summarizes our experiments and Sect. 6 concludes the paper.

2 RBAC, ARBAC, and the Reachability Problem

In *Role-Based Access Control (RBAC)*, access decisions are based on the roles that individual users have as part of an organization. The process of defining roles is based on a careful analysis of how an organization operates. Permissions are grouped by role name and correspond to various uses of a resource. A permission is restricted to individuals authorized to assume the associated role and represents a unit of control, subject to regulatory constraints within the RBAC model. For example, within a hospital, the role of doctor can include operations to perform diagnosis, prescribe medication, and order laboratory tests; the role of nurse can be limited to a strict subset of the permissions assigned to a doctor such as order laboratory tests.



User Assignment (UA)

User	Role
Alice	PCMember
Bob	Faculty
Charlie	Faculty
David	TA
David	Student
Eve	UEmployee
Fred	Student
Greg	UMember

Permission Assignment (PA)

Role	Permission
PCMember	GrantTenure
PCMember	AssignGrades
Faculty	AssignGrades
Faculty	ReceiveHBenefits
Faculty	UseGym
TA	AssignHWScores
TA	UseGym
UEmployee	ReceiveHBenefits
UEmployee	UseGym
Student	Register4Courses
Student	UseGym
UMember	UseGym

Fig. 1. User and Permission Assignments; and Role Hierarchies

We formalize a *RBAC policy* as a tuple $(U, R, P, UA,\ PA, \succeq)$ where U is a set of users, R a set of roles, and P a set of permissions. A binary relation $UA \subseteq U \times R$ represents a user-role assignment and a binary relation $PA \subseteq R \times P$ represents a role-permission assignment. A user-role assignment specifies the roles to which the user has been assigned while a role-permission assignment specifies the permissions that have been granted to a role. A partial order $\succeq$ on R is a role hierarchy of the policy, where $r_1 \succeq r_2$ means that r_1 is *more senior than* r_2 for $r_1, r_2 \in R$, i.e., every permission assigned to r_2 is also available to r_1.

A user u is an *explicit member* of role r when $(u, r) \in UA$ while the user u is an *implicit member* of role r if there exists $r' \in R$ such that $r' \succeq r$ and $(u, r') \in UA$. A user u *has permission* p if there exists a role $r \in R$ such that $(r, p) \in PA$ and u is a (explicit or implicit) member of r.

Example 1. Consider an RBAC policy describing a department in a university as depicted in Fig. 1. The top-left table is the user-role assignment, the top-right is the role-permission assignment, and the bottom is an example of role hierarchies (The role at the tail of an arrow is more senior than the one at the head).

Let us consider the user *Charlie*: he is an *explicit* member of role *Faculty* because the tuple (*Charlie*, *Faculty*) is in the user-role assignment *UA*. Additionally, role *Faculty* has been assigned to permissions *AssignGrades*, *ReceiveHBenefits*, and *UseGym*. Thus, *Charlie* can assign grades, receive benefits and use the gym through the role *Faculty*.

Let us consider the role hierarchy: role *Faculty* is more *senior* than role *UEmployee* (i.e., *Faculty* $\succeq$ *UEmployee*). Therefore, *Charlie* is an *implicit* member of the role *UEmployee*, and thus he can also use all permissions assigned to the role *UEmployee*. □

2.1 Administrative RBAC (ARBAC)

Access control policies need to be maintained according to the evolving needs of the organization. For flexibility and scalability in large distributed systems, several administrators are usually required and there is a need not only to have a consistent policy but also to ensure that the policy is modified by administrators who are allowed to do so.

Several administrative frameworks have been proposed on top of the RBAC model to address these issues. One of the most popular administrative frameworks is Administrative RBAC (ARBAC) [6] that controls how RBAC policies may evolve through administrative actions that update the UA and PA relations (e.g., actions that update UA include assigning or revoking user memberships into roles).

Formalization. Usually, administrators may only update the relation UA while PA and $\succeq$ are assumed constant. This is because a change in PA and/or $\succeq$ implies a change in the organization (see [16] for more detail). From now on, we focus on situations where U and R are finite, P plays no role, and $\succeq$ can be ignored[1] (and then, we only need to process the explicit members of a role when considering the role member relations). Thus, a RBAC policy is a tuple (U, R, UA) or for short UA if U and R are clear from the context.

Since administrators can be only partially trusted, administration privileges must be limited to selected parts of the RBAC policies, called *administrative domains*. An administrative domain is specified by a *pre-condition* defined as follows:

Definition 1. *A* pre-condition *C is a finite set of expressions of the forms r or $\overline{r}$ where $r \in R$.*

A user $u \in U$ *satisfies* a pre-condition C if, for each $\ell \in C$, u is a member of r when ℓ is r or u is not a member of r when ℓ is $\overline{r}$ for $r \in R$. We also say that r is a positive role and $\overline{r}$ is a negative role in C.

Permission to assign users to roles is specified by a ternary relation *can_assign* containing tuples of the form (C_a, C, r) where C_a and C are pre-conditions, and r a role. Permission to revoke users from roles is specified by a binary relation *can_revoke* containing tuples of the form (C_a, r) where C_a is a pre-condition and r a role. In both cases, we say that C_a is the *administrative pre-condition*, C is a *(simple) pre-condition*, r is the *target role*, and a user u_a satisfying C_a is the *administrator*. The relation *can_revoke* is only binary because simple pre-conditions are useless when revoking roles (see, e.g., [16]). When there exist users satisfying the administrative and the simple (if the case) pre-conditions of an administrative action, the action is *enabled*.

The semantics of the administrative actions in the ARBAC policy $\psi := (\mathit{can\_assign}, \mathit{can\_revoke})$ is given by the binary relation $\rightarrow_\psi$ defined as follows:

[1] We can transform a policy with role hierarchies to a policy without them by preprocessing away the role hierarchies as shown in [15].

Definition 2. *$UA \rightarrow_\psi UA'$ iff there exist users u_a and u in U such that either:*

- *there exists $(C_a, C, r) \in$ can_assign, u_a satisfies C_a, u satisfies C (i.e. (C_a, C, r) is enabled), and $UA' = UA \cup \{(u, r)\}$ or*
- *there exists $(C_a, r) \in$ can_revoke, u_a satisfies C_a (i.e. (C_a, r) is enabled), and $UA' = UA \backslash \{(u, r)\}$.*

A *run* of the administrative actions in $\psi := (can\_assign, can\_revoke)$ is a possibly infinite sequence $UA_0, UA_1, ..., UA_n, ...$ such that $UA_i \rightarrow_\psi UA_{i+1}$ for $i \geq 0$.

Example 2. Consider the RBAC policy with the UA relation depicted in Fig. 1 and an administrative action $(\{PCMember\}, \{Student, \overline{TA}\}, PTEmpl) \in$ *can_assign*, i.e., the administrative pre-condition is $C_a = \{PCMember\}$, the simple pre-condition is $C = \{Student, \overline{TA}\}$, and the target role is *PTEmpl*.

User *Alice* satisfies the pre-condition C_a because $(Alice, PCMember) \in UA$. User *Fred* satisfies the pre-condition C because he is a *Student* but not a *TA* (e.g., $(Fred, Student) \in UA$ and $(Fred, TA) \notin UA$). As a sequence, the administrative action is enabled.

We can update the current UA to $UA' = UA \cup \{(Fred, PTEmpl)\}$ by executing the following instance of the administrative action specified above: administrator *Alice* (who has role *PCMember*) assigns role *PTEmpl* to user *Fred.*

Notice that *Alice* cannot assign role *PTEmpl* to *David* because he is not only a *Student* but also a *TA* (i.e., *David* does not satisfy the pre-condition C). □

2.2 The User-Role Reachability Problem

Normally, policy designers and administrators want to foresee if the interactions among administrative actions, as seen in the Example 2, can lead the system to conflict states violating the security requirements of the organization (e.g., the security requirements forbid a user to be assigned to some sensitive roles). Thus, they need to analyze access control policies in order to discover such violation. This problem is called as the user-role reachability problem and is defined as follows.

Definition 3. *A pair (u_g, R_g) is called a* (RBAC) goal *for $u_g \in U$ and R_g a finite set of roles. The cardinality $|R_g|$ of R_g is the* size *of the goal.*

Definition 4. *Given an initial RBAC policy UA_0, a goal (u_g, R_g), and administrative actions $\psi = (can\_assign, can\_revoke)$; (an instance of) the* **user-role reachability problem***, identified by the tuple $\langle UA, \psi, (u_g, R_g)\rangle$, consists of checking if there exists a finite sequence $UA_0, UA_1, ..., UA_n$ (for $n \geq 0$) where (i) $UA_i \rightarrow_\psi UA_{i+1}$ for each $i = 0, ..., n-1$ and (ii) u_g is a member of each role of R_g in UA_n.*

In real scenario, subtle interactions between administrative actions in real policies may arise that are difficult to be foreseen by policy designers and administrators. Thus, automated analysis techniques are thus of paramount importance to analyze such policies and answer the user-role reachability problem.

The analysis techniques we will present in the following will be able to establish this automatically for the problem in ARBAC.

3 Model Checking Modulo Theories and the Reachability Problem

Model Checking Modulo Theories (MCMT). MCMT [9] is a framework to solve reachability problems for infinite state systems that can be represented by transition systems whose set of states and transitions are encoded as constraints in first-order logic. Several systems have been abstracted using such symbolic transition system such as parametrised protocols, sequential programs manipulating arrays, timed system, etc. (see again [9] for an overview).

MCMT framework uses a backward reachability procedure that repeatedly computes the so-called pre-images of the set of *goal* states, that is usually obtained by complementing a certain safety property that the system should satisfy. Then, the set of backward reachable states of the system is obtained by taking the union of the pre-images. At each iteration of the procedure, the procedure checks whether the intersection between the set of backward reachable states and the initial set of states is non-empty (i.e., *safety* test) or not (i.e., the *unsafety* of the system: there exists a (finite) sequence of transitions that leads the system from an initial state to one satisfying the goal). Otherwise, when the intersection is empty, the procedure checks if the set of backward reachable states is contained in the set computed at the previous iteration (*fix-point* test) and, if yes, the *safety* of the system (i.e. no (finite) sequence of transitions leads the system from an initial state to one satisfying the goal) is returned. Since sets of states and transitions are represented by first-order constraints, the computation of pre-images reduces to simple symbolic manipulations and testing safety and fix-point to solving a particular class of constraint satisfiability problems, called Satisfiability Modulo Theories (SMT) problems, for which scalable and efficient SMT solvers are currently available (e.g., Z3 [2]).

ASASPXL. In [3,5], it is studied how the MCMT approach can be used to solve (variants of) the user-role reachability problem. On the theoretical side, it is shown that the backward reachability procedure described above decides (variants of) the user-role reachability problem. On the practical side, extensive experiments have shown that an automated tool, called ASASP [4] implementing (a refinement of) the backward reachability procedure, has a good *trade-off* between *scalability* and *expressiveness*. Immediately after ASASP, a set of much larger instances of the user-role reachability problem has been considered in [10]. Unfortunately, ASASP does not scale to solve the set of problem. This is in line with the following observation of [10]: "model checking does not scale adequately for verifying policies of very large sizes." Then, in [13], a new tool based on the MCMT approach, called ASASPXL, has been proposed to efficiently solve much larger instances of the user-role reachability problem. The new analysis tool ASASPXL is build on top of MCMT, the first implementation of the MCMT approach. The choice of building a new analysis tool instead of modifying ASASP

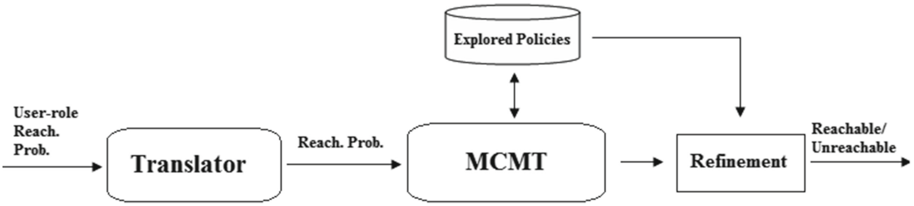


Fig. 2. ASASPXL architecture

gives some advantage. First, we only need to write a translator from instances of the user-role reachability problem to reachability problems in MCMT input language, a routine programming task. Second, MCMT has been developed and extensively used for the past years. It is thus more robust and offers a high degree of confidence. Third, we can re-use some features of a better engineered incarnation of the MCMT approach that can be exploited to significantly improve performances, as shown in [13].

The structure of ASASPXL is depicted in Fig. 2. It takes as input an instance of the user-role reachability problem and returns `reachable`, when there exists a finite sequence of administrative operations that leads from the initial RBAC policy to one satisfying the goal, and `unreachable` otherwise. To give such results, ASASPXL firstly translates the user-role reachability problem to the reachability problem in MCMT input language (module **Translator**). Then, it calls the model checker MCMT to verify the reachability of the problem. Finally, according to the answer returned by the model checker (in the data storage **Explored Policies**), ASASPXL refines it and returns `reachable` or `unreachable` as its output (module **Refinement**).

To keep technicalities to a minimum, we illustrate the translation on an instance of the user-role reachability problem as follows.

Example 3. Let $U = \{u_1, u_2, u_3, u_4, u_5\}$, $R = \{r_1, ..., r_8\}$, initially $UA := \{(u_1, r_1), (u_2, r_2), (u_5, r_5)\}$, and

$$(\{r_1\}, \{r_2\}, r_3) \in can\_assign \tag{1}$$

$$(\{r_3\}, \{r_4, \overline{r_5}\}, r_6) \in can\_assign \tag{2}$$

$$(\{r_4\}, \{r_5\}, r_7) \in can\_assign \tag{3}$$

$$(\{r_2\}, \{r_7\}, r_8) \in can\_assign \tag{4}$$

$$(\{r_2\}, r_3) \in can\_revoke \tag{5}$$

$$(\{r_5\}, r_4) \in can\_revoke \tag{6}$$

The goal of the problem is $(u_5, \{r_8\})$.

To formalize this problem instance in MCMT, ASASPXL firstly generates an unary relation u_r per role $r \in R$. The initial relation UA can thus be expressed as

$$\forall x. \left[\begin{array}{l}(u_{r_1}(x) \leftrightarrow x = u_1) \wedge (u_{r_2}(x) \leftrightarrow x = u_2) \wedge (u_{r_5}(x) \leftrightarrow x = u_5) \wedge \neg u_{r_2}(x) \wedge \neg u_{r_3}(x) \wedge \\ \neg u_{r_4}(x) \wedge \neg u_{r_6}(x) \wedge \neg u_{r_7}(x) \wedge \neg u_{r_8}(x)\end{array}\right].$$

A tuple, for instance, $(\{r_3\}, \{r_4, \overline{r_5}\}, r_6)$ in *can_assign* is formalized as

$$\exists x \exists y. \left[u_{r_3}(x) \wedge u_{r_4}(y) \wedge \neg u_{r_5}(y) \wedge \forall \lambda.(u'_{r_6}(\lambda) \leftrightarrow (\lambda = y \vee u_{r_6}(\lambda))) \right]$$

and a tuple, for example, $(\{r_2\}, r_3)$ in *can_revoke* can be expressed as

$$\exists x \exists y. \left[u_{r_2}(x) \wedge u_{r_3}(y) \wedge \forall \lambda.(u'_{r_3}(\lambda) \leftrightarrow (\lambda \neq y \wedge u_{r_3}(\lambda))) \right]$$

where u_r and u'_r indicate the value of U_r immediately before and after, respectively, the execution of the administrative action (we also have omitted—for the sake of compactness—identical updates, i.e. a conjunct $\forall \lambda.(u'_r(\lambda) \leftrightarrow u_r(\lambda))$ for each role r distinct from the target role in the tuple of *can_assign* or *can_revoke*). The other administrative actions are translated in a similar way.

The goal $(u_5, \{r_8\})$ can be represented as:

$$\exists x. u_{r_8}(x) \wedge x = u_5$$

The pre-image of the goal, that is computed by the model checker MCMT, with respect to $(\{r_2\}, \{r_7\}, r_8)$ is the set of states from which it is possible to reach the goal by using the administrative action $(\{r_2\}, \{r_7\}, r_8)$. This is formalized as the formula (see [5] for details)

$$\exists x \exists y.((u_{r_7}(y) \wedge y = u_5) \wedge u_{r_2}(x)),$$

On this problem, MCMT returns `unreachable` (i.e., there does not exist a finite sequence of administrative operations that lead from the initial policy UA to one satisfying the goal). □

Recently, a tool named VAC has been proposed in [8] for solving the user-role reachability problem of ARBAC policies. In [8], it is shown that VAC outperforms RBAC-PAT [16], MOHAWK [10], and ASASPXL on the problems in [16] and on a new set of complex instances of the user-role reachability problem. It was natural to run ASASPXL on these new benchmark problems: rather disappointingly, it could tackle such problem instances, however, its performance cannot be comparable with the new tool VAC (e.g., ASASPXL returns time-out in some problems). The reason of the bad scalability of ASASPXL is that ASASPXL does not work well on the user-role reachability problems with some specific features such as the problem containing some sub-problems having same structure of administrative actions; and the problems in which no state can be reached from the initial state. These and other problems have lead us to design new heuristics to make ASASPXL more scalable, as we will see in the next sections.

4 ASASPXL with new Heuristics

To enable ASASPXL to scale up to analyze the complex instances of the user-role reachability problem as shown in the previous section, our main idea is to design heuristics that help to alleviate the so-called state explosion problem, one of the commonly known problems in model checking techniques that must be addressed to solve most real-world problems. One of the main source of complexity is the large number of administrative actions; thus, for scalability, the original set of actions must be refined by using heuristics that tries to eliminate administrative actions that do not contribute to the analysis of RBAC policy. This and other techniques to control the state explosion problem will be detailed in the following (sub-)sections. Before going to the details of heuristics, we emphasize that all heuristics in the following will be implemented in a module named **Heuristics** and will be put before module **Translator** in the architecture of ASASPXL in Fig. 2. The ASASPXL's input, a user-role reachability problem, will be processed by module **Heuristics** before being forwarded to module **Translator** and then to module MCMT as described in Sect. 3.

4.1 Backward Useful Actions

The main idea to alleviate the state explosion problem is to eliminate as much as possible administrative actions that is useless to the analysis of ARBAC policy. This is done by extracting increasingly larger sub-sets of the tuples in the original set of administrative actions ψ so as to generate a sequence of increasingly more precise approximations of the original instance of the user-role reachability problem. The heuristics to do this is based on the following notion of an administrative action being useful.

Definition 5. Let ψ be the set of administrative actions and R_g a set of roles in an ARBAC policy:

- A tuple in ψ is 0-*useful* iff its target role is in R_g.
- A tuple in ψ is k-*useful* (for $k > 0$) iff it is $(k-1)$-useful or its target role occurs (possibly negated) in **either** the simple pre-condition **or** the administrative pre-condition of a $(k-1)$-useful transition.

A tuple t in ψ is *useful* iff there exists $k \geq 0$ such that t is k-useful.

Let $\psi^{\leq k} = (\mathit{can\_assign}^{\leq k}, \mathit{can\_revoke}^{\leq k})$ denote the set of all k-useful tuples in $\psi = (\mathit{can\_assign}, \mathit{can\_revoke})$. It is easy to see that $\mathit{can\_assign}^{\leq k} \subseteq \mathit{can\_assign}^{\leq k+1}$ and $\mathit{can\_revoke}^{\leq k} \subseteq \mathit{can\_revoke}^{\leq k+1}$ (abbreviated by $\psi^{\leq k} \subseteq \psi^{\leq k+1}$) for $k \geq 0$. Since the sets $\mathit{can\_assign}$ and $\mathit{can\_revoke}$ in ψ are bounded, there must exist a value $\tilde{k} \geq 0$ such that $\psi^{\leq \tilde{k}} = \psi^{\leq \tilde{k}+1}$ (that abbreviates $\psi^{\leq \tilde{k}} \subseteq \psi^{\leq \tilde{k}+1}$ and $\psi^{\leq \tilde{k}+1} \subseteq \psi^{\leq \tilde{k}}$) or, equivalently, $\psi^{\leq \tilde{k}}$ is the (least) fix-point, also denoted with $\mathit{lfp}(\psi)$, of useful tuples in ψ. Indeed, a tuple in ψ is useful iff it is in $\mathit{lfp}(\psi)$.

Example 4. Let ψ be the administrative actions in Example 3 and $R_g := \{r_8\}$. The sets of k-useful tuples for $k \geq 0$ are the following:

$$\begin{aligned} \psi^{\leq 0} &:= (\{(\{r_2\}, \{r_7\}, r_8)\}, \emptyset) \\ \psi^{\leq 1} &:= \psi^{\leq 0} \cup (\{(\{r_4\}, \{r_5\}, r_7)\}, \emptyset) \\ \psi^{\leq 2} &:= \psi^{\leq 1} \cup (\emptyset, \{(\{r_5\}, r_4)\}) \\ \psi^{\leq k} &:= \psi^{\leq 2} \text{ for } k > 2 \end{aligned}$$

Now, we run the user-role reachability problem: $\langle UA, \psi^{\leq 2}, (u_5, \{r_8\})\rangle$. ASASPXL returns `unreachable` on this problem instance. We obtain the same result if we run the tool on the translation of the following problem instance: $\langle UA, \psi, (u_5, \{r_8\})\rangle$. This leads to the following proposition. □

Proposition 1. A goal (u_g, R_g) is unreachable from an initial user-role assignment relation UA by using the administrative operations in ψ iff (u_g, R_g) is unreachable from UA by using the administrative operations in $lfp(\psi)$.

The proof of this fact can be obtained by slightly adapting the proof for the proposition in [13] and is thus omitted here.

4.2 Forward Useful Actions

In Sect. 4.1, we have introduced a heuristics identifying the set of useful actions (that is a subset of the original set of administrative actions) that is enough for solving the user-role reachability. We start using the roles in the goal to identify 0-useful actions and then using roles in the pre-conditions of k-useful actions to decide $(k+1)$-useful actions. Dually, we can start from the roles in the initial states and *forwardly* compute the set of useful actions. This is captured by the notion of *forward* useful action as follows:

Definition 6. Let ψ be the set of administrative actions, R be the set of roles, and $R_i := \{r | (u, r) \in UA_0\} \cup \{\overline{r} | r \in R\}$ be a set of roles occurring in the initial policy UA_0. A tuple $\tau \in \psi$:

- is forward 0-*useful* iff its pre-condition is a subset of R_i
- is forward k-*useful* (for $k > 0$) iff it is:
  - $(k-1)$-useful or,
  - its pre-condition is a subset of $R_i = R_i \cup \{r|$ r is the target role of a $(k-1)$-useful action$\}$

τ is *forward useful* iff there exists $k \geq 0$ such that τ is forward k-useful.

Let $\psi_F^{\leq k} = (can\_assign^{\leq k}, can\_revoke^{\leq k})$ denote the set of forward k-useful actions in $\psi = (can\_assign, can\_revoke)$, it is easy to see that $\psi_F^{\leq k} \subseteq \psi_F^{\leq k+1}$ for $k \geq 0$ and there exists a value $\tilde{k} \geq 0$ such that $\psi_F^{\leq \tilde{k}} = \psi_F^{\leq \tilde{k}+1}$ (i.e., $lfp_F(\psi) = \psi_F^{\leq \tilde{k}}$). Similar to the heuristic for backward useful actions above, we conclude the following proposition.

Proposition 2. A goal (u_g, R_g) is unreachable from an initial user-role assignment relation UA by using the administrative operations in ψ iff (u_g, R_g) is unreachable from UA by using the administrative operations in $\mathit{lfp}_F(\psi)$.

Example 5. Let consider again Example 3. The set R_i of roles in UA_0 is $\{r_1, r_2, r_5, \overline{r_1}, \overline{r_2}, ..., \overline{r_7}, \overline{r_8}\}$.

The sets of forward k-useful tuples for $k \geq 0$ are the following:

$$\begin{aligned}\psi_F^{\leq 0} &:= \{(\{r_1\}, \{r_2\}, r_3), (\{r_2\}, r_3), (\{r_5\}, r_4),\\ \psi_F^{\leq k} &:= \psi_F^{\leq 0} \text{ for } k > 0,\end{aligned}$$

ASASPXL returns `unreachable` on the user-role reachability problem $\langle \mathit{UA}, \psi_F^{\leq 0}, (u_1, \{r_8\})\rangle$ that confirms the results in Examples 3 and 4. □

The Combination of Backward and Forward Useful Actions. The module **Heuristics** in Sect. 4 works as follows to take into consideration the forward and backward useful actions. First, the module computes ψ^k and ψ_F^k that are the set of backward k-useful and forward k-useful actions, respectively. Then, the module will compute the intersection ψ_U of the sets ψ^k and ψ_F^k that is expected to be much smaller than ψ^k, ψ_F^k, and the original set ψ. Finally, the set of useful actions ψ_U is used to replace the original set ψ in solving the user-role reachability problem. The correctness and completeness of taking into consideration the intersection instead of the set of forward or backward useful actions is guaranteed by Proposition 3 that is simply a corollary of Propositions 1 and 2.

Proposition 3. A goal (u_g, R_g) is unreachable from an initial user-role assignment relation UA by using the administrative actions in ψ iff (u_g, R_g) is unreachable from UA by using the administrative operations in $\mathit{lfp}(\psi) \cap \mathit{lfp}_F(\psi)$.

4.3 Ordering Administrative Actions

We recall that the module MCMT implements the backward reachability procedure that computes the sets of backward reachable states from the goal. Basically, at each iteration, the procedure takes the first administrative action in the set ψ, computes its backward reachable states (pre-image) and then checks the intersection between the initial state and the backward states (by using an SMT solver to check the satisfiability). If the intersection is not empty (i.e., the goal is reachable from the initial state), the procedure returns `reachable` and stops. Otherwise, it selects the second action and repeats the process until all actions have been considered. This idea gives two advantages: first, the procedure can stop as soon as possible when it decides that the goal is reachable by checking an action and thus, not necessary to check the remaining actions; second, the fix-point formula can be divided into a set of smaller formulae, namely *local fix-points*, that is easier to be checked by SMT solvers. The original fix-point is reached when all the local fix-points are reached.

Clearly, the selection of the next action for computing the pre-images should be handled carefully since this will cause some redundant in the analysis that may negatively affect the performances of the procedure. In fact, if the goal is reachable and the administrative action, let us say τ, that helps the procedure in deciding the reachability of the goal is at the end of the action list, the current version of the backward reachability procedure must computes the pre-images for all actions before τ that are actually redundant computations. It is thus desirable to design a heuristics to select the next action to maximize the possibility of picking up an action that is important to show the reachability of the goal.

Our heuristics is based on the idea of how "close" between the set of states produced by computing the pre-image with respect to a given action and the set of initial states. This is because for each iteration, the procedure checks if the intersection between the pre-image generated by the given action and the set of initial states is empty, and then uses this check to decide the reachability of the goal. To illustrate how an action is "closer" than another, let us consider the following example:

Example 6. Let $U = \{u_1, u_2\}$, $R = \{r_a, r_1, ..., r_7\}$ initially $UA := \{(u_1, r_a), (u_1, r_1), (u_1, r_2), (u_1, r_5)\}$, and the set ψ contains:

$$(\{r_a\}, \{r_1, r_2, \overline{r_4}\}, r_7) \in \mathit{can\_assign} \tag{7}$$

$$(\{r_a\}, \{r_1, r_3\}, r_7) \in \mathit{can\_assign} \tag{8}$$

The pre-images of the two actions (7) and (8) (computed by the backward reachability procedure) are represented by formulae $\exists x, y.(r_a(x) \wedge r_1(y) \wedge r_2(y) \wedge \neg r_4(y))$ and $\exists x, y.(r_a(x) \wedge r_1(y) \wedge r_3(y))$, respectively. It is easy to see that the set of reachable states of action (7) is contained in the initial state UA (i.e., their intersection is not empty). We also notice how all the roles in the precondition of action (7) appear in UA while role r_3 in the precondition of action (8) does not. In this case, we say that action (7) is closer (to the initial state) than action (8). Then, action (7) should be selected before action (8) in the backward reachability procedure. □

We define the function *Diff* calculating how "close" two sets of roles are as follows:

Definition 7. Let C_1 and C_2 be pre-conditions, the difference between C_1 and C_2 is:

$$\mathit{Diff}(C_1, C_2) := (C_1^+ \backslash C_2^+) \cup (C_1^- \backslash C_2^-)$$

where C_1^+ and C_2^+ are sets of positive roles in C_1 and C_2, respectively; C_1^- and C_2^- are sets of negative roles in C_1 and C_2, respectively.

We illustrate how the function *Diff* is used in the heuristic by the following example:

Example 7. Let us consider again Example 6. First, the heuristic will calculate $R_i = \{r_a, r_1, r_2, r_5, \overline{r_a}, \overline{r_1}, ..., \overline{r_7}\}$ that represents all roles occurring in the initial UA as defined in Definition 6.

Let consider action (7) with its precondition $C_1 = \{r_a, r_1, r_2, \overline{r_4}\}$, the heuristic then computes $\mathit{Diff}(C_1, R_i) = \emptyset$. Similarly, the precondition of action (8) is $C_2 = \{r_a, r_1, r_3\}$ and $\mathit{Diff}(C_2, R_i) = \{r_3\}$.

Since $|\mathit{Diff}(C_2, R_i)| > |\mathit{Diff}(C_1, R_i)|$, we say that action (7) is closer (to the initial state) than action (8). In other words, the precondition C_1 can be easily satisfied by the initial UA while C_2 requires more tuples, for instance $(u_1, r_3) \in UA$, to be satisfied. Thus, the heuristic will select the actions (7) to compute its pre-image before (8) □

We add this heuristic to the tool ASASPXL by adding a sub-module, namely **Ordering the Actions**, to module **Heuristics** mentioned above. After computing the set of useful actions as in Sects. 4.1 and 4.2, **Heuristics** will invoke the sub-module **Ordering the Actions** with the set ψ_U of useful actions as the parameter. The sub-module then orders the administrative actions in ψ_U and returns the ordered set as workflow below:

1. Let ψ_U be the set of actions and R_i containing all roles occurring in the initial state UA_0.
2. For each $\tau = (C_a, C, r) \in \psi_U$:
   (a) If $C_a = \emptyset$ and $C = \emptyset$:
      i. set τ be the first order in ψ_U (for several actions with $C_a = C = \emptyset$, we do not care the order between them)
   (b) Else:
      i. Calculate $\mathit{Diff}_\tau := \mathit{Diff}(C_a \cup C, R_i)$ for τ
3. Order the actions in ψ_U by their $|\mathit{Diff}_\tau|$ (from lower value to higher one)
   (a) If $|\mathit{Diff}_{\tau_1}| = |\mathit{Diff}_{\tau 2}|$ where $\tau_1 = (C_{a1}, C_1, r_1)$ and $\tau_2 = (C_{a2}, C_2, r_2)$:
      i. τ_1 has higher order if $|C_{a1} \cup C_1| < |C_{a2} \cup C_2|$ and vice versa

Initially, the procedure computes the set R_i containing all roles in the initial UA_0. Then, it calculates the set Diff for each administrative action in ψ_U (Step 2). Administrative actions of the form $(\emptyset, \emptyset, r)$ are set highest order in Step 2(a) since its pre-conditions are alway satisfied. For several actions of the form $(\emptyset, \emptyset, r)$, we do not care about the order between them. The procedure then classifies the actions in ψ_U based on their Diff (Step 3). Notice how the procedure prioritizes the action containing smaller set of pre-conditions (Step 3(a)) for the actions having the same $|\mathit{Diff}|$. This is because the formula representing the set of backward reachable states generated by the action (see, e.g., Example 6) may be smaller (i.e., containing less literals) than the others and thus easier for the SMT solver to check the satisfiability.

5 Experiments

We have implemented ASASPXL and heuristics in Python and used the MCMT model checker [1] for computing the pre-images. We have also conducted an experimental evaluation to show the scalibility of asaspXL and compare it with state-of-the-art analysis tools such as Mohawk [10], Vac [8], and Pms [17] on two benchmark sets from [10] and [8]. Note that Pms contains 2 versions, namely *Prl* and *Fwd* that implement the analysis with/without applying their parallel algorithm [17].

Remark. Sometimes, to simplify the analysis of ARBAC policies, *separate administration assumption* (for short, SA) has been applied (see, e.g. [16]) which amounts to requiring that administrative roles (i.e., roles occurring in the administrative precondition C_a) and regular roles (i.e., roles occurring in the simple precondition C) are disjoint. This permits to consider just one user, omit administrative users and roles so that the tuples in *can_assign* are pairs composed of a simple precondition and a target role (i.e., (C, r)) and the pairs in *can_revoke* reduce to target roles only (i.e., (r)). In the state-of-the-art analysis tools mentioned above, Mohawk requires this assumption while the other two and asaspXL do not need it. The benchmarks are thus classified as either SA benchmarks (that require SA assumption) or non-SA benchmarks (that do not the the assumption) as in the following.

Description of Benchmarks. The first benchmark set is a SA benchmark taken from [10]. It contains three synthetic test suites: **Test suite 1** contains policies in which roles occur only positively in the (simple) pre-conditions of *can_assign* rules and the set of *can_revoke* rules is non-empty. **Test suite 2** contains policies in which roles occur both positively and negatively in *can_assign* rules and the set of *can_revoke* rules is empty. **Test suite 3** contains policies in which roles occur both positively and negatively in *can_assign* rules and the set of *can_revoke* rules is non-empty. The second benchmark set is a non-SA benchmark from [17]. It contains 10 instances of the user-role reachability problem inspired by a university.

Evaluation. We perform all the experiments on an Intel Core I5 (2.6 GHz) CPU with 4 GB Ram running Ubuntu 11.10

Table 1 reports the results of running asaspXL, Pms, Vac and Mohawk on the first benchmark set. Notice that all problems in this benchmark are unsafe (i.e., analysis tools returns "reachable"). Column 1 shows the name of the test suite, column 2 contains the number of roles and administrative operations in the policy. Columns 3, 4, 6 and 7, and 8 show the average times (in seconds) taken by Mohawk, Vac, Pms (with two versions), and asaspXL, respectively, to solve the instances of the user-role reachability problem associated to an ARBAC policy. For Mohawk and Vac, the average time also include the time spent in the slicing phase (a technique for eliminating irrelevant users, roles, and administrative operations that are non relevant to solve a certain instance of the user-role reachability problem, see [8,10] for more details) and the verification phase. Column 6 and 10 represent the number of actions remaining after the slicing phase of Vac and the useful actions obtained by asaspXL, respectively.

Experiments for the benchmark that does not adopt the separate administration assumption are reported in Tables 2; their columns have the same

Table 1. Experimental results on the "complex" benchmarks in [10]

(Separate administration assumption)								
Test suite	# Roles ◇	Mohawk	Vac		Pms		asaspXL	
					Fwd	*Prll*		
	#Rules	Time	Time	# Rules	Time	Time	Time	# Rules
Test suite 1	3 ◇ 15	0.45	0.29	1	0.38	0.45	**0.09**	1
	5 ◇ 25	0.53	0.35	1	0.38	0.47	**0.11**	1
	20 ◇ 100	0.64	0.35	1	0.35	0.39	**0.12**	1
	40 ◇ 200	0.97	0.69	1	0.49	0.57	**0.31**	2
	200 ◇ 1000	2.69	0.95	1	0.47	0.55	**0.38**	1
	500 ◇ 2500	4.88	1.59	1	0.97	1.16	**0.70**	1
	4000 ◇ 20000	16.99	1.88	1	33.55	22.39	**1.27**	2
	20000 ◇ 80000	51.57	2.72	1	*TO*	*TO*	**1.27**	2
	30000 ◇ 120000	65.51	4.12	1	*TO*	*TO*	**1.69**	2
	40000 ◇ 200000	131.17	9.94	1	*TO*	*TO*	**2.29**	2
Test suite 2	3 ◇ 15	0.45	0.25	1	0.36	0.37	**0.15**	1
	5 ◇ 25	0.55	0.39	1	0.35	0.38	**0.28**	1
	20 ◇ 100	0.59	0.24	1	0.32	0.49	**0.16**	1
	40 ◇ 200	1.21	0.56	1	0.54	0.59	**0.15**	1
	200 ◇ 1000	2.55	0.83	1	0.59	0.63	**0.14**	1
	500 ◇ 2500	6.12	1.52	1	1.54	0.83	**0.47**	2
	4000 ◇ 20000	15.51	1.63	1	29.17	21.39	**1.18**	2
	20000 ◇ 80000	26.12	5.25	1	*TO*	*TO*	**1.22**	2
	30000 ◇ 120000	98.95	6.73	1	*TO*	*TO*	**1.28**	2
	40000 ◇ 200000	146.84	11.89	1	*TO*	*TO*	**1.43**	2
Test suite 3	3 ◇ 15	0.51	0.15	1	0.37	0.35	**0.08**	1
	5 ◇ 25	0.45	0.19	1	0.55	0.49	**0.09**	1
	20 ◇ 100	0.87	0.31	1	0.42	0.62	**0.16**	1
	40 ◇ 200	0.99	0.67	1	0.46	0.57	**0.19**	2
	200 ◇ 1000	7.23	2.12	1	0.92	1.28	**0.56**	2
	500, 2500	4.69	1.20	1	0.74	0.97	**0.10**	1
	4000 ◇ 20000	15.15	4.61	1	20.49	15.13	**1.17**	2
	20000 ◇ 80000	32.35	3.85	1	*TO*	*TO*	**2.25**	2
	30000 ◇ 120000	115.11	9.65	1	*TO*	*TO*	**1.69**	2
	40000 ◇ 200000	157.35	10.32	1	*TO*	*TO*	**2.55**	2

TO: time out *Err*: Error *m*: minute

Table 2. Experimental results on the benchmarks in [17]

(Non separate administration assumption)								
Test case	# Roles ◇ # Rules	Answer	Vac		Pms		asaspXL	
					Fwd	*Prll*		
			Time	# Rules	Time	Time	Time	# Rules
Test 1	40 ◇ 487	Unsafe	17.25	3	0.83	**0.68**	1.15	2
Test 2	40 ◇ 450	Safe	0.21	0	0.91	0.75	**0.19**	0
Test 3	40 ◇ 462	Unsafe	9.33	3	0.92	0.93	**0.71**	2
Test 4	40 ◇ 446	Unsafe	7.51	3	0.99	45.16	**0.69**	2
Test 5	40 ◇ 480	Unsafe	48.31	47	1.25	**0.91**	2.12	9
Test 6	40 ◇ 479	Unsafe	26.62	13	1.02	**0.86**	1.69	4
Test 7	40 ◇ 467	Unsafe	1 m 12.56	101	4.22	3.26	**1.85**	2
Test 8	40 ◇ 484	Unsafe	1 m 16.23	65	5.08	2 m 16.21	**2.04**	8
Test 9	40 ◇ 463	Unsafe	1 m 35.11	89	5.91	6 m 35.24	**2.91**	11
Test 10	40 ◇ 481	Unsafe	29.94	38	**0.65**	0.75	2.45	5

semantics as in previous table with additional column "Answer" reports the results returned by analysis tools (*Safe* means the goal is unreachable while *Unsafe* means the goal is reachable). We do not report the experimental result of Mohawk because it cannot handle user-role reachability problems without the separate administration assumption.

The results clearly show that asaspXL performs significantly better than Mohawk, Pms, and Vac in the first benchmark set (Table 1). Notice that Pms throws a time-out (that is set to 10 min) in the biggest test cases. For the second benchmark set, asaspXL outperforms Pms and is much better than Vac. We emphasize that the number of actions after using module **Heuristics** in asaspXL is reduced significantly (column 9).

Table 3. Experimental results when turning on/off heuristics in Sect. 4

Test case	# Roles ◇ # Rules	Answer	asaspXL	
			Without Heuristic	**With** heuristics
Test 1	40 ◇ 487	Unsafe	2 m 52.73	**1.15**
Test 2	40 ◇ 450	Safe	16.22	**0.19**
Test 3	40 ◇ 462	Unsafe	1 m 1.63	**0.71**
Test 4	40 ◇ 446	Unsafe	57.15	**0.69**
Test 5	40 ◇ 480	Unsafe	2 m 35.87	**2.12**
Test 6	40 ◇ 479	Unsafe	2 m 45.71	**1.69**
Test 7	40 ◇ 467	Unsafe	3 m 17.33	**1.85**
Test 8	40 ◇ 484	Unsafe	*TO*	**2.04**
Test 9	40 ◇ 463	Unsafe	*TO*	**2.91**

TO: time out *Err*: Error *m*: minute

Table 3 shows experimental results when we run asaspXL on the instances of user-role reachability problem in Table 2 with/without heuristics introduced in Sect. 4. Columns 1, 2, and 3 have the same semantic as previous tables. Column 4 reports the analysis time when turning off heuristics while column 5 shows the performance obtained by using heuristics. The results prove the effectiveness of heuristics on the analysis. In many cases, the analysis time is reduced significantly, for example, from 3 min to nearly 2 s.

6 Conclusions

We have presented techniques to enable the MCMT approach to solve instances of user-role reachability problem. We have also designed a set of heuristics that help our analysis techniques to be more scalable. The main idea is to reduce as much as possible the number of administrative actions in the original problem. An excerpt of an exhaustive experimental evaluation has been conducted and provided evidence that an implementation of the proposed techniques and heuristics, called asaspXL, performs significantly better than mohawk, Vac, and Pms on a variety of benchmarks from [8,10].

As future work, we plan to design new heuristics based on some functionalities provided by the model checker mcmt such as the capability of tracking the visited states for later use. Another interesting line of research for future work is to consider the combination of backward and forward reachability procedure to speed up the analysis of the model checker.

References

1. http://homes.di.unimi.it/~ghilardi/mcmt
2. http://research.microsoft.com/en-us/um/redmond/projects/z3
3. Alberti, F., Armando, A., Ranise, S.: Efficient symbolic automated analysis of administrative role-based access control policies. In: Proceeding of ASIACCS, pp. 165–175. ACM Press (2011)
4. Alberti, F., Armando, A., Ranise, S.: ASASP: automated symbolic analysis of security policies. In: Bjørner, N., Sofronie-Stokkermans, V. (eds.) CADE 2011. LNCS (LNAI), vol. 6803, pp. 26–33. Springer, Heidelberg (2011). doi:10.1007/978-3-642-22438-6_4
5. Armando, A., Ranise, S.: Automated symbolic analysis of ARBAC policies. In: Cuellar, J., Lopez, J., Barthe, G., Pretschner, A. (eds.) STM 2010. LNCS, vol. 6710, pp. 17–34. Springer, Heidelberg (2011). doi:10.1007/978-3-642-22444-7_2
6. Crampton, J.: Understanding and developing role-based administrative models. In: Proceedings of 12th CCS, pp. 158–167. ACM Press (2005)
7. Capitani, D., di Vimercati, S., Foresti, S., Jajodia, S., Samarati, P.: Access control policies and languages. Int. J. Comput. Sci. Eng. (IJCSE) **3**(2), 94–102 (2007)
8. Ferrara, A.L., Madhusudan, P., Nguyen, T.L., Parlato, G.: Vac - verifier of administrative role-based access control policies. In: Biere, A., Bloem, R. (eds.) CAV 2014. LNCS, vol. 8559, pp. 184–191. Springer, Heidelberg (2014). doi:10.1007/978-3-319-08867-9_12

9. Ghilardi, S., Ranise, S.: Backward reachability of array-based systems by SMT solving: termination and invariant synthesis. Logical Methods Comput. Sci. (LMCS) **6**(4), 1–48 (2010)
10. Jayaraman, K., Ganesh, V., Tripunitara, M., Rinard, M., Chapin, S.: Automatic error finding for access control policies. In: Proceedings of 18th CCS, pp. 163–174. ACM (2011)
11. Jha, S., Li, N., Tripunitara, M.V., Wang, Q., Winsborough, H.: Towards Formal Verification of Role-Based Access Control Policies. IEEE Trans. Dependable Secure Comput. **5**(4), 242–255 (2008). IEEE
12. Li, N., Tripunitara, M.V.: Security analysis in role-based access control. ACM Trans. Inf. Syst. Secur. (TISSEC) **9**(4), 391–420 (2006). ACM Press
13. Ranise, S., Truong, A., Armando, A.: Boosting model checking to analyse large ARBAC policies. In: Jøsang, A., Samarati, P., Petrocchi, M. (eds.) STM 2012. LNCS, vol. 7783, pp. 273–288. Springer, Heidelberg (2013). doi:10.1007/978-3-642-38004-4_18
14. Sandhu, R., Coyne, E., Feinstein, H., Youmann, C.: Role-based access control models. IEEE Comput. **2**(29), 38–47 (1996). IEEE
15. Sasturkar, A., Yang, P., Stoller, S.D., Ramakrishnan, C.: Policy analysis for administrative role-based access control. Theor. Comput. Sci. **412**(44), 6208–6234 (2011). Elsevier
16. Stoller, S.D., Yang, P., Ramakrishnan, C., Gofman, M.I.: Efficient policy analysis for administrative role-based access control. In: Proceedings of 14th CCS, pp. 445–455. ACM Press (2007)
17. Yang, P., Gofman, M., Yang, Z.: Policy analysis for administrative role based access control without separate administration. In: Wang, L., Shafiq, B. (eds.) DBSec 2013. LNCS, vol. 7964, pp. 49–64. Springer, Heidelberg (2013). doi:10.1007/978-3-642-39256-6_4

Trust and Risk-Based Access Control for Privacy Preserving Threat Detection Systems

Nadia Metoui[1,2](✉), Michele Bezzi[3], and Alessandro Armando[1,4]

[1] Security and Trust Unit, FBK-Irst, Trento, Italy
nadia.metoui@gmail.com
[2] DISI, University of Trento, Trento, Italy
[3] SAP Labs France, Security Research, Sophia-Antipolis, France
[4] DIBRIS, University of Genova, Genoa, Italy

Abstract. Intrusion and threat detection systems analyze large amount of security-related data logs for detecting potentially harmful patterns. However, log data often contain sensitive and personal information, and their access and processing should be minimized. Anonymization can provide the technical mean to reduce the privacy risk, but it should carefully applied and balanced with utility requirements of the different phases of the process: a first exploration analysis needs less details than an investigation on a suspect set of logs. As a result, a complex access control framework has to be put in place to, simultaneously, address privacy and utility requirements. In this paper we propose a trust- and risk-aware access control framework for Threat Detection Systems, where each access request is evaluated by comparing the privacy-risk and the trustworthiness of the request. When the risk is too large compared to the trust level, the framework can apply adaptive adjustment strategies to decrease the risk (e.g., by selectively obfuscating the data) or to increase the trust level to perform a given task. We show how this model can provide meaningful results, and real-time performance, for an industrial threat detection solution.

Keywords: Trust · Risk · Privacy · Utility · Privacy-preserving threat detection

1 Introduction

Big Data analytics for security, based on the correlation of security events from several log files, play a key role in state-of-the-art threat detection and prevention techniques [25,33]. Threat detection systems, as intrusion detection systems, are typically characterized by an automatic pattern or anomaly detection phase, which can highlight suspicious events, followed by a detailed investigation performed by an human expert to decide if a real attack is detected or it is a false positive. In this phase, the expert often inspects the raw data (log files) triggering the alert.

However, log data often contain sensitive and personal information (e.g., user ids, IP addresses), and, although the security investigation can constitute a

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 285–304, 2016.
DOI: 10.1007/978-3-319-48057-2_20

legitimate purpose for their processing, the access and usage should be limited to the relevant and necessary data to accomplish a specific analysis.

Anonymization is often used to pre-process the data, removing sensitive information from log files, and enabling further processing with minimal privacy risk. However, this is achieved by deteriorating the quality or utility of the data. Although some analytics can still be run on anonymized log data [20], in many cases the anonymization can impact the quality of results, and, ultimately, decrease the ability to detect and react to cyber threats.

We propose a trust- and risk-aware access control framework for Threat Detection Systems (TDS), which addresses the concerns described above. Our framework does not require an *a priori*, i.e. off-line, anonymization of the data sources. The automatic pattern detection phase uses the original dataset and anonymization is applied only if further, human based, analysis is needed on the resulting data.

The risk level of each data request is dynamically evaluated by the access control decision point based on several parameters (e.g., context, role and trustworthiness of the requester), and, if needed, anonymization is applied on the specific resulting data set. In this study, we focus on *re-identification* risk and, following common practice, we use k-anonymity as risk metrics. However, our approach is not bound to these choices and can be adapted to use alternative metrics (e.g., l-diversity, t-closeness, and differential privacy). To summarize, the approach has multiple advantages:

- it limits the impact on the utility, since we apply the anonymization only after running the pattern detection on the original data, and we adapt the anonymization strategy to the specific pattern.
- it provides a simple framework to address the, often conflicting, privacy and utility requirements.
- it is based on concepts as trust and risk, which have an intuitive meaning in the business world.
- it offers a flexible configuration, allowing to define a trade-off between security and privacy suitable to the organization's priorities (risk and trust levels can be tuned to set a permissive or a restrictive access, the adjustment strategy can be configured to optimize utility or performance priorities, etc.)

To evaluate the effectiveness of the proposed approach, we have developed a prototype implementation and we experimentally evaluated it by running a number of threat detection patterns based on the SAP Enterprise Threat Detection (ETD) solution. The results obtained (reported in Sect. 4.6) show that the model can address the utility and performance requirements of a realistic use-case.

Structure of the Paper. In the next Section we provide a threat detection system use case, which we use to illustrate the main features of our risk-aware privacy preserving approach. In Sect. 3 we present a trust- and risk-aware approach for privacy enhancing access control model and we describe its application on the proposed use case Sect. 4 is dedicated to an experimental evaluation of the proposed approach in terms or performance, scalability and data utility (after anonymization). Lastly, we discuss the related work in Sect. 5 and we conclude in Sect. 6 with some final remarks.

2 Use Case

Modern intrusion detection systems at application level (called Threat Detection System, TDS, herein)[1], collect security information on the application stack and correlated it with context information, to detect potential threats.

A TDS, typically, first collects application level log files from various systems, it enriches the logs with contextual (e.g., time, location) information, and finally stores the events data in a database table.

The events data are then periodically automatically analyzed against predefined threat patterns to detect potential anomalies and attacks. Any matching with these patterns generates real time alerts. When an alert is raised a human user is informed and actions must be undertaken to evaluate and react to the alerts (e.g., investigate the validity of the alert or locate its cause). Figure 1 illustrates the architecture of the system, as well as the different users involved in the process.

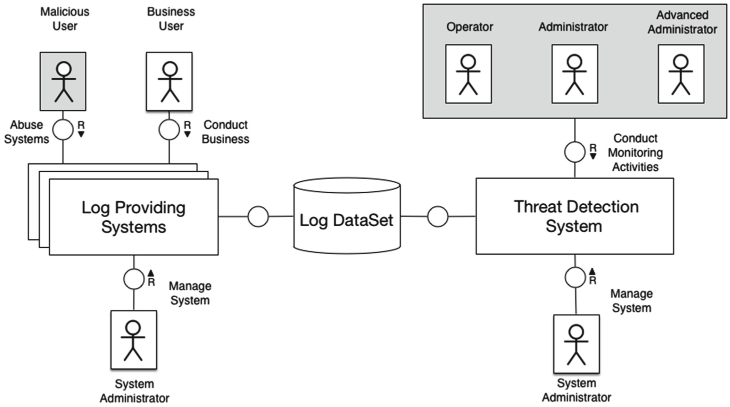


Fig. 1. Business roles and system landscape

For our purpose, two operations of the TDS are important:

- *Pattern detection.* A pattern is a representation of a combination of suspicious log events that could indicate a threat. It is often implemented as a set of filters applied to the event database, and compared with some thresholds.

[1] We refer to these systems as TDS, to distinguish them from network level intrusion detection systems (often called IDS or SIEM). Moreover, we base our description on the SAP Enterprise Threat Detection, but the analysis could be applied to other solutions, including IDS. For a comparison between application and network level intrusion detection systems, see [18].

Table 1. Roles

Operator	Classify alerts and report patterns anomalies His/Her tasks requires access to pattern detection results (events/log data related to the suspicious pattern) in case of alerts
Administrator	Has all *Operator* tasks and privileges. They can also Investigate alerts, Create or Reconfigure patterns. He/She should have access the detection results and events data related to the patterns
Advanced administrator	Has all *Administrator* tasks and privileges. Can also grant exceptional access to the data by attributing higher trust level to an *Operator* or an *Administrator*

If this threshold is exceeded an alert is triggered. For instance the ensemble of events indicating a *Failed Login* initiated by the same source (e.g., Terminal) may indicate a *Brute Force Attack* if the number of attempts exceeds, say, 20 attempts in less than 10 min.

- *Investigating Alert.* In this phase, an human operator investigates the alert, to decide if this is an actual attack or a false positive. It may require access to the details of the events triggering the alert, or at least of some attributes of these events.

The Investigation phase implies that TDS *Users* access some detailed information from the logs, we will provide some examples in Sect. 4.3. These *Users* have different functions within the process of monitoring potential threats, investigating them and reacting. Table 1 gives an example of how you can divide the user roles in the TDS, and the corresponding access authorizations required to execute their tasks.

Log files contain personal information, such user names, IP addresses, etc., and despite the security investigation can constitute a legitimate purpose for their processing, it should be done according to the data minimization principle, reducing the access to personal data. Therefore, TDS systems often perform some (pseudo-)anonymization before analyzing the event data, such as replacing real user name or IDs with pseudonyms.

However, with the increasing variety and complexity of collected log files, a full anonymization of the log dataset before processing could, on one hand, provide a good privacy protection, but also significantly impact the performance of the system, both in terms of the *utility* (the quality of results of the pattern detection phase, or the information available to the operator for the manual inspection) and processing time (anonymization on large data set could be time consuming, and on data stream re-run regularly).

To address this challenge, a more dynamic approach is needed: instead of anonymizing the complete event data base beforehand, whenever an user performs an operation accessing event tables, we have to apply specific anonymization methods which reduce the privacy risk, but preserving the most relevant information for that operation. In practice, the anonymization process should be customized for each operation (to preserve the information useful for completing the task) and for each type of users, which can have different level of access to the data. In the next section, we will propose a framework that to realize this scenario.

3 Privacy-Enhancing Risk-Based Access Control

In this section we provide a general description of our Trust- and Risk-Based Access Control model, based on previous model we introduced in [1], and we explain how it can be adapted to the use case described in Sect. 2.

3.1 Trust and Risk-Based Access Control

The framework evaluates access decisions using the trust and risk values related to the request. This access evaluation can be represented by the function $Auth(obj, u, p)$ defined as follows. User u is granted permission p on object obj iff the trustworthiness of the incoming request is larger or equal to the risk, i.e.,

$$Auth(obj, u, p) = \begin{cases} \mathbf{allow} & \text{if } T(u, C) - R(obj, p, C) \geq 0 \\ \mathbf{adjust}_{\Sigma}(T, R) & \text{otherwise} \end{cases} \tag{1}$$

where $T(u, C)$ is the trustworthiness of the request, which depends on user u and context information C (e.g., security emergency) and $R(obj, p, C)$ is the risk, which depends on the requested object obj (e.g. a table a file.) and the permission p (e.g., read or write)[2] and context C.

Access request in evaluated by comparing the risk of the access to the trustworthiness, which plays the role of risk threshold (in practice, the maximum amount of risk that an user can take in a certain context): If $T \geq R$ access is **allowed**, vice-versa if $T < R$, the access cannot be granted *as is*. However, risk-based access control models have been originally devised to increase information accessibility, and they tend to be more permissive (still keeping risk under control) than traditional access control systems. Along this reasoning, in case of $T < R$ instead of denying access, the system can propose an *adjustment strategy* $\sigma \in \Sigma$, to reach the condition $T \geq R$. Clearly, there are two possible methods for adjustment strategies: *(i)* Risk mitigation, σ_R, (decreasing R), or *(ii)* Trust enhancement strategies, σ_T, increase the trustworthiness T. Risk mitigation strategies can include anonymizing the data, or imposing additional obligations on data handling, whereas trust enhancement could be implemented by (temporary) privilege escalation or provision of additional credentials [1] However each of these strategies is expected to have some *negative* side effects: for example, anonymization degrades data quality, impacting utility or privilege escalation can increase the complexity of the security governance;

[2] In most cases the dependency of risk from permission is mediated by roles. For the sake of simplicity, we do not consider here roles, for an extension of this model including roles, we can follow the lines of the models described in [6].

accordingly, the choice of the optimal strategy should balance the access control objectives with the impact of the adjustment strategies.

If we focus on data access and privacy risk (as the use case in Sect. 2), and limiting the adjustment strategies to anonymization, we should find an optimal anonymization strategy $\hat{\sigma_R}$ among all the possible anonymization strategies Σ_R, which allows for data access limiting risk (so fulfilling Eq. 1), and, at the same time, maximizing the utility, after the strategy σ_R is applied: $U(\sigma_R)$. This is can be expressed as classical utility-privacy optimization problem:

$$\hat{\sigma_R} = \arg \max_{\sigma_R \in \Sigma_R} U_{\sigma_R(obj)} \tag{2}$$

$$s.t. \quad R_{\sigma_R} \leq T \tag{3}$$

In practical cases (as we will see in Sect. 3.2), the number of mitigation strategies can be very limited, and the optimization problem is reduced to testing a small set of anonymization strategies, and estimating either based on numerical thresholds or expert assessment, if the utility is sufficient for the business task. If this is not the case, trust enhancement mechanism can be triggered or access is denied. In the next subsections we will show how trust and risk can be modeled, with a focus on the application to Threat Detection Systems.

3.2 Privacy Enhancing Approach

Risk Model: Risk in IT security is generally expressed in terms of the likelihood of occurrence of certain (negative) events times the impact [14]. In this paper we will deal with the privacy breach risk. Privacy breaches are often associated with the concept of *individual identifiability*, used in most data protection privacy laws (e.g., EU data protection directive [13], Health Insurance Portability and Accountability Act (HIPAA) [27]). To prevent *individual identifiability* the regulation requires that disclosed information (alone or in combination with reasonably available information from other sources or auxiliary informations [24]) should not allow an intruder: to identify individuals in a dataset (identity disclosure) or to learn private/sensitive information about individuals (attribute disclosure) with a very high probability or confidence (see [29,32]).

To assess the privacy risk (when releasing a given dataset) various privacy metrics have been proposed in the literature (see [4,10] for a review). The most popular metric is k-anonymity [26][3].

In the k-anonymity approach attributes (or columns) in a dataset are classified as follows:

- *Identifiers:* Attributes that can uniquely identify individuals e.g., full name, social security number passport number.
- *Quasi-identifiers (QIs) or key attributes* Attributes that, when combined, can be used to identify an individual, e.g., age, job function, postal code

[3] Other privacy metrics exist (for example, ℓ-diversity, and t-closeness, see [15]), but k-anonymity is still a *de-facto* standard in real applications.

- *Sensitive attributes:* Attributes that contain intrinsically sensitive information about an individual, e.g., diseases, political or religious views, income.

In presence of identifiers the re-identification risk is clearly maximum (i.e., probability of re-identification $P = 1$), but even if identifiers are removed, combining QIs individuals can be singled out and this implies a high risk. k-anonymity condition requires that *every* combination of QIs is shared by at least k records in the dataset. A large k value indicates that the dataset has a low re-identification risk, because, at best, an attacker has a probability $P = 1/k$ to re-identify a data entry (i.e., associate the sensitive attribute of a record to the identity of a User). Therefore the (re-identification) risk related to a k-anonymous data-view v is:

$$risk(v) = 1/k_v \times I \tag{4}$$

where I is the impact. In most cases any identity disclosure is considered equally important, and, thus for simplicity sake we will set the impact $I = 1$ this will allow us to normalize the risk and the trust values to $[0, 1]$ (for a discussion on the impact normalization, see [1]).

Trust Model: Several definitions have been proposed in for the concept of Trust in the literature [17]. In this paper, trust plays the role of risk threshold: a very trusted user is allowed to take a large risk (for a discussion on how relating this definition with more classical trust metrics, see [1]. We assign trust level $T_{user}(u)$ to the users according to their competence/roles and the tasks this role is expected to fulfill (see Table 1). Following data minimization policy, a role should have *enough* trust to access the resources (data) needed to fulfill these task and not more. These values are assigned on a scale from 0 to 1, where 0 means that basically no privacy risk can be taken, therefore impacting significantly the quality of accessible data; and 1 means the role should be granted access to maximum amount of data.

Note: the same request can be used to fulfill different tasks in different contexts for instance *"Perform Maintenance and Improvement tasks"* or *"React to a Security Incident"* (if an alert is raised). In the latter the need to react to a security threat overcomes the privacy requirements and the request should receive more permissive results thus have higher level of trust we will define the two context-related trust levels as $T_{context}(Alert) = 1$ and $T_{context}(noAlert) = 0$

To compute the request trustworthiness (total trust value) we can use the approach for multi-dimensions trust computation proposed in [22], where the total trust is computed as weighted sum of trust factor values.

$$T = \sum_{i=1}^{n} W_i \times T_i(\beta_i) \tag{5}$$

with $\{\beta_1 ... \beta_n\}$ a set of trust factors and $T_i()$ and W_i respectively the trust function and weight of the i^{th} trust factors, with $\sum_{i=1}^{n} W_i = 1$ and $T \in [0, 1]$

We are in a 2-dimensions trust case thus we will express our total trust value as the following

$$T(q) = W \times T_{user}(u) + (1 - W) \times T_{context}(c) \tag{6}$$

Adjustment Strategies:

Risk Mitigation: A possible way to decrease the disclosure risk is anonymization. Anonymization is a commonly used practice to reduce privacy risk, consisting in obfuscating, in part or completely, the personal identifiable information in a dataset. Anonymization methods include [9]:

- *Suppression:* Removal of certain records or part of these records (columns, tuples, etc., such UserId column);
- *Generalization:* Recoding data into broader classes (e.g., releasing only a Network prefixes instead of IP addresses etc.) or by rounding/clustering numerical data;

Traditionally, anonymization is run off-line, but more recently risk-based access control models, which use in-the-fly anonymization as mitigation strategy have been proposed [2].

Trust Enhancement: Trust enhancement mechanisms can realized by asking the user to provide additional guarantees (i.e., additional credential) or proofs of obligation enforcement. In our case, we may require trust enhancement for an emergency alert, where there is the need to increase the access to the original data for investigation. This could be implemented as a change in the context, which impacts the trust value according to Eq. 6, or simply increasing temporarily the trust of an user T_u (privilege escalation).

4 Experimental Evaluation

We validate our approach by applying the described model to the scenario described in Sect. 2. The threat detection system is expected to provide real time and a accurate results. In this section of the paper we will investigate the impacts our approach has on the functioning of the threat detection system and whether the expected *Performance* and *Utility* matches the requirement of a real-time.

More in details, as mentioned in Sect. 2 the threat detection system allows to automatically detect potential attack patterns, and then, if an additional investigation is needed, a human operator can browse the log data of the events corresponding to certain pattern for manual inspection.

Ideally, the Operator should be in the position to perform the manual analysis, so to decide if the detected pattern is a false or true positive, on data where the personal information are anonymized (or in any case, where the re-identification risk is low). In fact, if the operator has not sufficient information to decide, they needs to access less anonymized (more risky) data, or in other words to get higher access privileges (trust enhancement) getting Administrator rights, or directly involving an Administrator.

Accordingly, we need to check:

- *Utility.* Does the model allow a low trusted operator (i.e., small risk threshold) to perform the investigation in most cases, and relying on trust enhancement for the remaining cases?
- *Performance.* Does the additional anonymization step impact real-time performance?

Before addressing these questions (see Sect. 4.6), we need to describe our prototype implementation (Sect. 4.1), the data set and its classification from a privacy risk perspective (Sect. 4.2), the selection of typical patterns used for the validation (Sect. 4.3), the utility measure (Sect. 4.5) and the trust level setting (Sect. 4.4).

4.1 Prototype Implementation

We developed a prototype of our framework, based on the implementation described in [3]. Our prototype is implemented in Java 8 and uses SAP HANA Database. It is composed from 3 main modules:

- The *Risk Aware Access Control module:* mimics a typical XACML data flow, providing an implementation of the PDP, the PEP and the PIP functionality as well as a set of authorization policies.
- The *Risk Estimation module:* evaluates the privacy risk using pre-configured criteria (privacy metrics, anonymization technique, identifying information). It compares the privacy risk to the request trustworthiness level, then produces an estimation of the minimal anonymization to be applied in order to meet this level.
- The *Trust and Risk Adjustment module:* we implemented the Risk Adjustment Component to perform anonymization. It uses ARX [19] a Java anonymization framework implementing well established privacy anonymization algorithms and privacy metrics such as k-anonymity, ℓ-diversity, t-closeness, etc. (the Trust Adjustment Component was not implemented in this version of the prototype.)

4.2 Data Set and Privacy Classification

To test the performance of our framework in the TDS use case, we used a data set containing around 1 *bn* record of log data collected from real SAP systems deployed in test environment. The logs data set is composed 20 fields (in Table 2 we present a summery of the most important fields)

As described in Sect. 3.2, to anonymize a data set, we first need to formalize our assumptions on the attributes that can be use to re-identify the entry, or, in other words, classify the attributes in terms of identifiers, QIs and sensitive attributes. This classification, typically, depends on the specific domain.

Table 2. An extract of the Log dataset columns, privacy classification of each column and anonymization technique to be applied

Log events data set		
Attribute	Type	Anonymization
EventID	Non-sensitive	
Timestamp	Sensitive	
UserId (Origin)	Identifier	Suppression
UserId (Target)	Identifier	Suppression
SystemId (Origin)	QI	Generalization
SystemId (Target)	QI	Generalization
Hostname (Origin)	QI	Generalization
IPAddress (Origin)	QI	Truncation
MACAddress (Origin)	QI	Truncation
TransactionName	Sensitive	
TargetResource	Sensitive	

QIs should include the attributes a possible attacker is likely to have access to from other sources, whereas sensitive attributes depend on the application the anonymized data are used for. For example, in our experiments we set (obviously) User ID as an identifier, and the IP address as a quasi-identifier. Similarly, we assume that the Transaction name (the called function) cannot provide any help for re-identification, therefore we consider it a sensitive attribute (and no anonymization will be applied). Table 2 provides an example of this classification, and, for identifiers and quasi-identifiers, the corresponding anonymization methods applied.

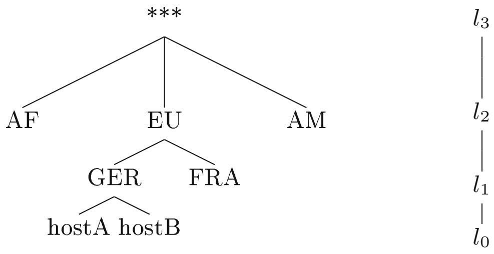


Fig. 2. The generalization hierarchy for host names is organized as following: l_1 and l_2 are a location based generalization by country then by continent. in level l_3 host names are totally obfuscated and entirely revealed at the level l_0.

4.3 Pattern Detection and Investigation

In our experiments we focus on 5 typically *Patterns* with different and increasing complexity in terms of the size of the returned views and the privacy risk. Two different kind of queries are used during each phase respectively *Detection*

Queries and *Investigation Queries*. The selected queries {**Q1** ... **Q5**} described in Table 3 are all *Investigation Queries*. An *Investigation Query* is a “SELECT *” extracting all the details of the events corresponding to certain pattern.

Table 3. Queries: resulting views size and risk level

Query	Corresponding pattern	View size	Risk level
Q1	Brute force attack	Large (50550)	Very high ($k = 2$)
Q2	Security configuration changed	Large (40300)	Medium ($k = 7$)
Q3	Blacklisted function called	Medium (14500)	Very high ($k = 1$)
Q4	Table dropped or altered	Small (228)	Medium ($k = 6$)
Q5	User assigned to admin group	Very small (12)	Very high ($k = 1$)

4.4 Roles and Trustworthiness Levels

We have 3 roles Operator, Administrator and an Advanced administrator with increasing access requirements (to fulfill their tasks), and we that expect to require increasing privacy clearance, or in other words,to be able to accept larger risk. Usually, for k-anonymity, k values in the range $3-10$ are considered medium risk, $k > 10$ low risk, and for $k \leq 2$ the risk is very high (clearly, for $k = 1$ the risk is the maximum, no anonymity) [11]. Therefore we propose the parameter setting described in Table 4, where for sake of simplicity we have considered a single trust factor $T = T_u$ (i.e. we set $W = 1$ in Eq. 6).

Table 4. Users/Roles Privacy clearances and Trustworthiness levels

Role	Access requirement	Privacy clearance	Trust level (risk threshold)
Operator	Low	Minimal ($k > 10$)	$T_u \in [0.05, 0.1]$
Admin	Medium	Medium ($k > 2$)	$T_u \in [0.1, 0.5]$
Adv. Admin.	High	Maximum ($k \leq 2$)	$T_u \in [0.5, 1]$

4.5 Utility Evaluation

The effect of anonymization terms of utility is a widely discussed issue in the literature several generic metrics have been proposed to quantify the *“damage”* caused by anonymization (see [16] for a review). However, these metrics do not make any assumption on the usage of the data (so called *syntactic metrics*), limiting their applicability on realistic use-cases.

Other approaches propose to assess the accuracy loss (Utility loss) of a system (i.e., IDS in [21], Classifier in [5]) by comparing the results of certain operations run on original then anonymized dataset using use case related criteria (i.e., in the context of a TDS the comparison criteria can be the number *False positives*)

Although interesting for our context, this approach can not be applied in our use case, since it assumes that the analysis is run directly on anonymized data, whereas, in our use case, the pattern detection is performed on *clear* data, and the anonymization is applied only on the results (data-view).

We propose a method combining both approaches and that would include an evaluation:

- *From Syntactic standpoint:* The information loss caused by the anonymization, we use the Precision Metric that allows us to estimate the precision degradation of QIs based on the level of generalization with respect to the generalization tree depth (e.g., for th generalization tree Fig. 2 if we allow access to continent instead of host-names we used the 3^{rd} level generalization out of 4 possible levels so $d_p(hostnames) = 3/4 = 75\,\%$ precision degradation for host-names).
- *From Functional standpoint:* The effect of this loss on our use case. During the investigation phase, the operator, mostly, bases their analysis on a sub-set of attributes, which are different for each attack pattern. Thus we will assign a utility coefficient uc to different attributes based on the relevance of the attribute to the pattern/query.

Combining the to approaches we compute the utility degradation of a data-view v as

$$U_d(v) = \sum_{a_i \in A} uc_{a_i} \times d_p(a_i) \tag{7}$$

with $A = \{a_1..a_i\}$ the set of attributes in the data set. We also set the precision degradation of the identifiers to $d_p(identifiers) = 1$ as they will be totally suppressed after the anonymization.

4.6 Results and Analysis

For our experiments, we want to investigate: *(i)* Performance: the impact of on-the-fly anonymization (as risk mitigation strategy) on the performance (response time). *(ii)* Utility: we would like to investigate if the quality of resulting data is generally enough to fulfill the expected tasks for every user/role for various pattern investigation.

In order to evaluate these aspects we run several experiments considering 5 patterns and 7 users/role with different trustworthiness level, $t = \{0.055, 0.083\}$ Operators, $t = \{0.12, 0.15, 0.45\}$ Administrators, and $t = \{0.9, 1\}$ Advanced Administrators. The corresponding size and anonymity level of the views returned by the queries (corresponding to the selected patterns) are reported in Table 3. In the rest of this section we will indicate both the queries and the corresponding views as **Q1**, **Q2**, **Q3**, **Q4** and **Q5**.

Performance and Scalability. To evaluate the the performance of our tool, including the computational overhead caused by the anonymization, we run queries **Q1**, **Q2**, **Q4**, and **Q5** (described in Table 3) using our access control prototype experiment, 100 times for each query to average out the variance of

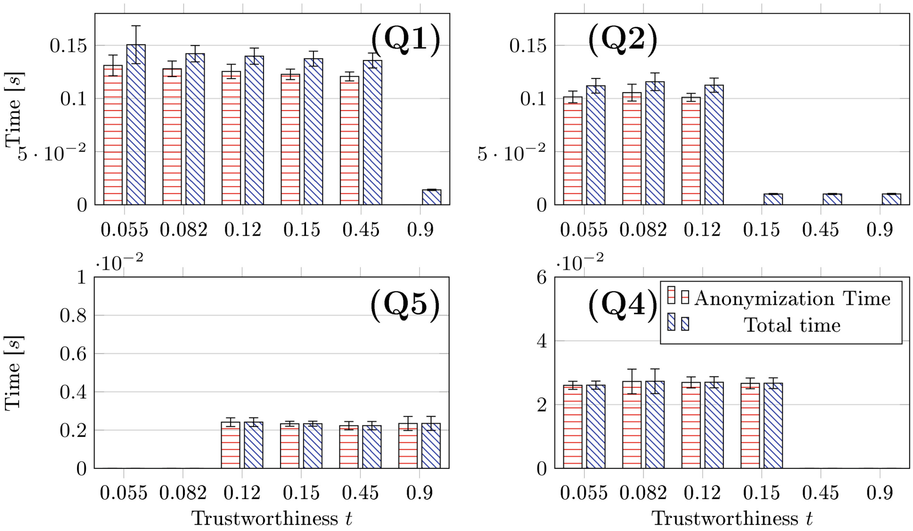


Fig. 3. Average anonymisation time (horizontal striped bars) and average total response time (diagonally striped bars) for **Q1**, **Q2**, **Q4**, and **Q5** (data-views) and 6 different users (trust levels).

the response time. In Fig. 3 we report the results of the experiments for the four queries for the 6 trustworthiness levels.

For **Q1**, we observe that the anonymization process increases significantly the response time. In fact when the query is carried out by the most trusted user ($t = 0.9$), with no anonymization needed, the response time on average is less then 15 ms (see Fig. 3.Q1, diagonally striped bar corresponding to $t = 0.9$). By decreasing the trustworthiness of the requester the view must be anonymized and the average response time increases to 150 ms in the worst case (cf. Fig. 3.Q1, diagonally striped bar corresponding to $t = 0.055$). This time difference is entirely due to the anonymization time (130 ms, as shown in Fig. 3, **Q1**, horizontal striped bars corresponding to $t = 0.055$). Increasing the trust level decreases the needed anonymization, but it slightly affects anonymization time. We can observe a similar behavior in the other queries (see Fig. 3, **Q2**, **Q4**, and **Q5**), with an increase of response time when anonymization takes place and no significant variations in performance for different levels of anonymization. For instance, for **Q2** and **Q4** we have two views with an already medium level of anonymity (respectively $k = 7$ and $k = 6$),the anonymization (when needed) still impacts the performance in the same scale then **Q1** and **Q5** with very low anonymity level (respectively $k = 2$ and $k = 1$).

From these experiments, we observe that when anonymization is applied the response time increases, but, even in the worst cases, the increase is far less than one order of magnitude, and, basically, it has no impact on the real-time response of the system. Moreover, the application of different levels of

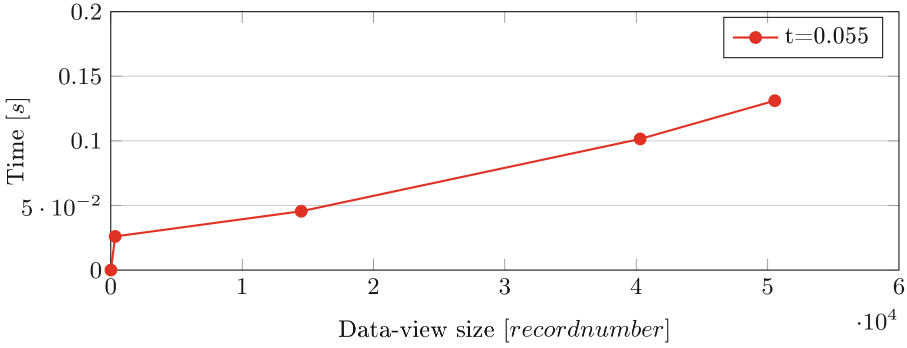


Fig. 4. Average anonymization time variation according to data-view sizes (for trustworthiness $t = 0.055$).

anonymization (different k in our case) have a small impact. We will investigate in the next paragraph the effect of the data-view size on the Anonymization and Response time.

Let us analyze the behavior of the anonymization time increasing the size of the data set. Typically patterns run in limited time window (e.g., 10 to 30 min) producing small-sized data-views (i.e., in the range of $10 - 10^3$). To investigate the scalability of our approach, in Fig. 4, we report the average anonymization time variation for 5 different data-view {**Q1** to **Q5**} (with 5 different sizes see Table 3) and a low trustworthiness level ($t = 0.055$, so anonymization is always applied). As mentioned above, the worst case (around 510^4 records) takes less than 150 ms, and a linear extrapolation of the data allows as to estimate the anonymization time for a 10^5 data view (so, 100 times the typical size) around 200 ms, which it can be safely considered as a real-time response for our use case.

Utility: Trustworthiness levels (i.e., risk threshold) should be set to allow the best a trade off between data exploitation and privacy protection. In our use case we set our trustworthiness levels respecting a conventional distribution of privacy risk levels presented in Table 4, and we would like to investigate the convenience of this repartition by answering the following question: Do these trustworthiness levels provide enough data (or data with enough utility) to allow each user/role to fulfill their tasks described in Table 1. In Fig. 5, we report the the utility degradation according the six selected trustworthiness levels, representing the 3 roles (reported on the top of the figure). We can observe that the utility degradation (obviously) decreases as we increase the trust level, with the limiting case of $t = 1$ with no utility loss (and no anonymization) for the Advanced Administrator. For most of the patterns (4 over 5, so except **Q5**), the Operator role has a maximum utility loss of 30 %, showing that the specific anonymization transformations applied are strongly decreasing the risk, and limiting the impact on the utility. That should allow to perform the analysis on the anonymized data, without the need to enhance the trust level (so no need to get Admin rights).

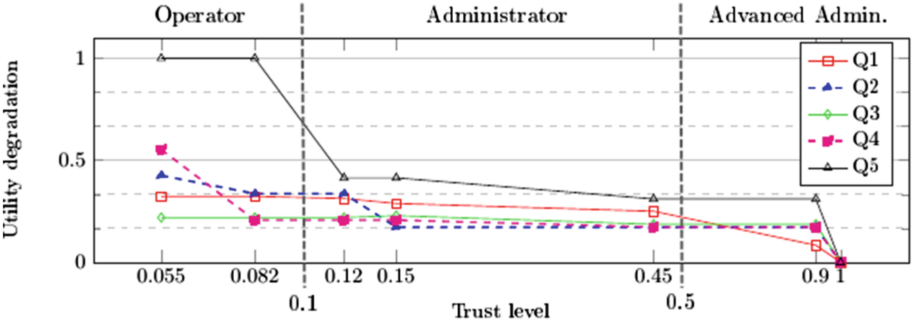


Fig. 5. Utility degradation by trust level for different queries

In the case of **Q5**, the anonymization is not able to significantly decreases the risk, without largely impacting the utility. In fact, the Operator is left with no information (utility degradation = 1), and to analyze the result an increase of the acceptable risk threshold (trust level) is needed. Enhancing trust (i.e. assigning Admin rights to the Operator) could reduce the utility degradation in the 30 %–40 % range, likely allowing the assessment of the pattern result. We should note, that **Q5** is particularly hard to anonymize, because it has a small amount of events (around 10), and, since k-anonymity is measure of indistinguishability, it needs strong anonymization.

Figure 5 also shows that in most cases increasing the trust level for Administrator or even Advanced Administrator (except of course for $t = 1$, where we have no anonymization) the impact on utility degradation is moderate: for example **Q1** and **Q4** are almost flat in the Administrator zone, similarly **Q2** has a first drop, and stays flat in the Administrator and Advanced Administrator parts. In other words, increasing the risk thresholds, we could take more risk, but we do not gain much in terms of the utility. This counter-intuitive effect is mostly due to the difficulty to find an anonymization strategy able to equalize the risk threshold. As mentioned in Sect. 3.1, in practical cases the number of possible anonymization strategies is limited, and to fulfill the condition of Eq. 1 the final risk may be quite below the risk thresholds (trust values). In practice, in many cases, even increasing the risk thresholds (trust values), it is not possible to find a more optimal (from the utility point of view) anonymization strategy. In Fig. 5 we show the utility loss for four patterns both showing the risk thresholds (dotted lines) and the *actual* risk achieved after the anonymization. In the ideal case, the two curves should be the same, meaning that we could always find a transformation that equalize actual risk and risk thresholds (trust), but in practice we see that we are often far from this optimal condition. For example, for pattern **Q2**, with risk thresholds $t = 0.15$, $t = 0.45$ (Administrator role) and $t = 0.9$ (Advanced Administrator), indicated with red circles, we have the same value of utility degradation. In fact, the anonymization strategy found for $t = 0.15$ case, corresponds to an actual risk of 0.14 (square dots with a circle in Fig. 5, upper-right panel), so quite close to the threshold. Increasing the thresholds to

$t = 0.45$ and $t = 0.9$ (round dots with a circle in the figure), no better strategies were found, so the same anonymization strategy is applied, and clearly the final risk is still 0.14 (and utility is the same), well below the thresholds. Similar effects are also present for the other patterns.

The experimental analysis shows that adapting the anonymization to the specific patterns, we can mostly preserve enough information for the investigation, keeping the privacy risk low. In cases where this is not sufficient, typically characterized by small data set, the trust enhancement strategy can support the access to less-anonymized data.

5 Related Work

Privacy Issues in Intrusion Detection: Privacy issues related to shearing and/or using network and log data in IDS and TDSs has received a growing interest in the last few years. Several analysis were proposed in the literature to describe privacy breaches related to sharing and using log data and privacy preserving approaches have been proposed to address these issues.

A strict enforcement of the *need-to-know* principle has been proposed for reducing the likelihood of privacy violations. For example, Ulltveit-Moe et *al.* in [31] propose to set two profiles of users according to the expertise level: the first profile allows monitoring tasks using anonymized data the second consists of security experts, with clearance to perform necessary privacy-sensitive operations to investigate attacks. This model clearly increases the privacy protection, but it is hard to apply in realistic cases, since it relies on anonymizing the entire (source) data set beforehand, resulting in either low privacy or low utility. In our approach, we use a similar approach, strictly adopting the need-to-know principle, but, as described in Sect. 3, the anonymization is dynamically only on the data set resulting from a pattern, and according to the *trust* level of the users/roles. As a result, we can use the *better* anonymization transformation depending on the specific utility of each pattern, assuring an increase of both privacy and utility.

Other works focus on specific anonymization techniques for logs (see [23] for review), and on measuring the privacy risk. For example, in [30], the authors use entropy to measure privacy leakage in IDS alerts. We implemented several of the proposed anonymization techniques in our prototype, and, although based on k-anonymity, our framework can include other privacy measures by changing the risk function. More specifically, entropy based privacy metrics can be easily integrated with k-anonymity approach, as shown in [20].

Risk Based Access Control Systems: Several risk and trust based access control models have been introduced in the last years. (e.g. [6–8, 12, 28]), where for each access request or permission activation, the corresponding risk is estimated and if the risk is less than a threshold (often related to trust) then the operation is permitted, otherwise it is denied. Cheng et al. [8] estimate risk and trust thresholds from the sensitivity labels of the resource and clearance level of the users in a multi-level-security system. They also consider a trust enhancement mechanism (the authors call it risk mitigation strategy in their paper) that allow users to spend *tokens* to access resources with risk higher than their trust level. The details on how this mechanism can be applied in real cases are not provided.

Chen et al. [6] introduced an abstract model which allows role activation based on a risk evaluation compared to predefined risk thresholds. Trust values are considered, and they impact (decreasing) risk calculation. If risk is too high, the model includes mitigation strategies, indicated as system obligations. The paper does not specify how to compute the risk thresholds, trust, and the structure of obligations. In a derived model [7], mitigation strategies have been explicitly defined in terms of user obligations (actions that have to be fulfilled by the user). The model also introduces the concept of *diligence score*, which measured the diligence of the user to fulfill the obligations (as a behavioral trust model), and impact the risk estimation. Another extension has been proposed [2,3], focusing on re-identification risk and anonymization is used as mitigation strategy (as in our paper).

Following the original Chen et al. [6] model, these papers consider trust as part of the risk value. We can essentially map our model to the Chen et al. [6] approach; in fact renaming the difference $R - T$ as risk in Eq. 1, and explicitly defining as a threshold the impact of risk mitigation, we obtain mostly the same model as described in [6]. However, as we discussed in [1], explicitly introducing the risk/trust comparison allows for: *(i)* trust enhancement and risk mitigation strategies are clearly separated, making easier to find an optimal set of strategies to increase access, keeping risk under control, *(ii)* trust thresholds are not dependent on the risk scenario, and, if we consider multiple risk factors, we can compare the overall risk with the trust. Our model addresses these issues, clearly separating trust aspects from risk.

6 Conclusions and Future work

Motivated by a strong need to improve privacy protection in security monitoring products, such as Threat Detection Systems, we proposed an access control model able to address their, complex, privacy and utility requirements. We adapted a Risk-based Access Control approach (described in [1,2]) for a threat detection solution, where anonymization is dynamically applied to reduce the privacy risk. Automatically applying specific anonymization strategies, in real-time, for each pattern, we showed how this model is able to provide a simple solution for investigating potentially harmful patterns, with a minimal privacy risk. In the cases where significantly reducing risk results in an excessive degradation of the quality of data, the model supports mechanisms of trust enhancement to access less-anonymized data. We also showed that the anonymization step does not impact the real-time performance of the systems for typical data set.

We based our analysis on real TDS, using a small sample of typical patterns. A more extensive analysis is needed to be able to implement a robust solution.

In particular, the parameter setting (risk thresholds) can be complex in presence of a large number of patterns. In addition, although widely used k-anonymity has its own limitation, for example, in presence of multiple overlapping data sets, it is well known that the k-anonymity condition cannot be fulfilled (lack of composability). Other privacy models exist, such differential privacy, which could be integrated in our framework.

Acknowledgments. The research leading to these results has received funding from the FP7 EU-funded project SECENTIS (FP7-PEOPLE-2012-ITN, grant no. 317387) and EU-funded project TOREADOR (contract n. H2020-688797).

References

1. Armando, A., Bezzi, M., Cerbo, F., Metoui, N.: Balancing trust and risk in access control. In: Debruyne, C., Panetto, H., Meersman, R., Dillon, T., Weichhart, G., An, Y., Ardagna, C.A. (eds.) OTM 2015. LNCS (ISAIH), vol. 9415, pp. 660–676. Springer, Heidelberg (2015). doi:10.1007/978-3-319-26148-5_45
2. Armando, A., Bezzi, M., Metoui, N., Sabetta, A.: Risk-aware information disclosure. In: Garcia-Alfaro, J., Herrera-Joancomartí, J., Lupu, E., Posegga, J., Aldini, A., Martinelli, F., Suri, N. (eds.) DPM/QASA/SETOP -2014. LNCS, vol. 8872, pp. 266–276. Springer, Heidelberg (2015). doi:10.1007/978-3-319-17016-9_17
3. Armando, A., Bezzi, M., Metoui, N., Sabetta, A.: Risk-based privacy-aware information disclosure. Int. J. Secur. Softw. Eng. **6**(2), 70–89 (2015). http://dx.doi.org/10.4018/IJSSE.2015040104
4. Bezzi, M.: An information theoretic approach for privacy metrics. Trans. Data Priv. **3**(3), 199–215 (2010)
5. Brickell, J., Shmatikov, V.: The cost of privacy: destruction of data-mining utility in anonymized data publishing. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008, pp. 70–78. ACM, NewYork (2008). http://doi.acm.org/10.1145/1401890.1401904
6. Chen, L., Crampton, J.: Risk-aware role-based access control. In: Meadows, C., Fernandez-Gago, C. (eds.) STM 2011. LNCS, vol. 7170, pp. 140–156. Springer, Heidelberg (2012). doi:10.1007/978-3-642-29963-6_11
7. Chen, L., Crampton, J., Kollingbaum, M.J., Norman, T.J.: Obligations in risk-aware access control. In: Cuppens-Boulahia, N., Fong, P., García-Alfaro, J., Marsh, S., Steghöfer, J. (eds.) PST, pp. 145–152. IEEE (2012). http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6287257
8. Cheng, P.C., Rohatgi, P., Keser, C., Karger, P.A., Wagner, G.M., Reninger, A.S.: Fuzzy multi-level security: an experiment on quantified risk-adaptive access control. In: IEEE Symposium on Security and Privacy, pp. 222–230. IEEE Computer Society (2007). http://dblp.uni-trier.de/db/conf/sp/sp.2007.html#ChengRKKWR07
9. Ciriani, V., De Capitani di Vimercati, S., Foresti, S., Samarati, P.: Theory of privacy and anonymity. In: Atallah, M., Blanton, M. (eds.) Algorithms and Theory of Computation Handbook, 2nd edn. CRC Press (2009)
10. Clifton, C., Tassa, T.: On syntactic anonymity and differential privacy. Trans. Data Priv. **6**(2), 161–183 (2013). http://dl.acm.org/citation.cfm?id=2612167.2612170
11. Committee on Strategies for Responsible Sharing of Clinical Trial Data: Sharing Clinical Trial Data: Maximizing Benefits, Minimizing Risk. National Academies Press (US), Washington (DC) (2015)

12. Dickens, L., Russo, A., Cheng, P.C., Lobo, J.: Towards learning risk estimation functions for access control. In: Snowbird Learning Workshop (2010). https://www.usukitacs.com/papers/6006/TA2_22_Dickens_learning_risk_estimation.pdf
13. FRA and the Council of Europe: handbook on european data protection law. Technical report (2014)
14. Friedewald, M., Pohoryles, R.J.: Privacy and Security in the Digital Age: Privacy in the Age of Super-Technologies. Routledge, Abingdon (2016)
15. Fung, B.C.M., Wang, K., Chen, R., Yu, P.S.: Privacy-preserving data publishing: a survey of recent developments. ACM Comput. Surv. **42**(4), 14:1–14:53 (2010). http://doi.acm.org/10.1145/1749603.1749605
16. Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N.: Fast data anonymization with low information loss. In: Proceedings of the 33rd International Conference on Very Large Data Bases, pp. 758–769 (2007). VLDB Endowment
17. Josang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems **43**(2), 618–644 (2007). Emerging issues in Collaborative Commerce. http://www.sciencedirect.com/science/article/B6V8S-4GJK82P-1/2/a9a6e96414fa04641c1d31a57989618d
18. Kaempfer, M.: (2015). http://scn.sap.com/community/security/blog/2015/03/04/sap-enterprise-threat-detection-and-siem-is-this-not-the-same
19. Kohlmayer, F., Prasser, F., Eckert, C., Kuhn, K.A.: A flexible approach to distributed data anonymization. J. Biomed. Inform. **50**, 62–76 (2014). Special issue on Informatics Methods in Medical Privacy
20. Kounine, A., Bezzi, M.: Assessing disclosure risk in anonymized datasets. In: Proceedings of the FloCon Workshop, January 2009
21. Lakkaraju, K., Slagell, A.: Evaluating the utility of anonymized network traces for intrusion detection. In: Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks, SecureComm 2008, pp. 17:1–17:8. ACM, NewYork (2008). http://doi.acm.org/10.1145/1460877.1460899
22. Li, X., Zhou, F., Yang, X.: A multi-dimensional trust evaluation model for large-scale p2p computing. J. Parallel Distrib. Comput. **71**(6), 837–847 (2011)
23. Mivule, K., Anderson, B.: A study of usability-aware network trace anonymization. In: Science and Information Conference (SAI), 2015, pp. 1293–1304. IEEE (2015)
24. Narayanan, A., Huey, J., Felten, E.W.: A precautionary approach to big data privacy. In: Gutwirth, S., Leenes, R., De Hert, P. (eds.) Data Protection on the Move, vol. 24, pp. 357–385. Springer, Dordrecht (2016)
25. Oprea, A., Li, Z., Yen, T.F., Chin, S.H., Alrwais, S.: Detection of early-stage enterprise infection by mining large-scale log data. In: 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 45–56. IEEE (2015)
26. Samarati, P.: Protecting respondents' identities in microdata release. IEEE Trans. Knowl. Data Eng. **13**(6), 1010–1027 (2001)
27. Scholl, M.A., Stine, K.M., Hash, J., Bowen, P., Johnson, L.A., Smith, C.D., Steinberg, D.I.: Spp. 800–66 rev. 1. an introductory resource guide for implementing the health insurance portability and accountability act (hipaa) security rule. Technical report (2008)
28. Shaikh, R.A., Adi, K., Logrippo, L.: Dynamic risk-based decision access control systems. Comput. Secur. **31**(4), 447–464 (2012)
29. Templ, M., Meindl, B., Kowarik, A.: Introduction to statistical disclosure control (sdc). Project: Relative to the testing of SDC algorithms and provision of practical SDC, data analysis OG (2013)

30. Ulltveit-Moe, N., Oleshchuk, V.A.: Measuring privacy leakage for IDS rules. CoRR abs/1308.5421. http://arxiv.org/abs/1308.5421(2013)
31. Ulltveit-Moe, N., Oleshchuk, V.A., Køien, G.M.: Location-aware mobile intrusion detection with enhanced privacy in a 5G context. Wireless Pers. Commun. **57**(3), 317–338 (2011)
32. Vaidya, J., Clifton, C.W., Zhu, Y.M.: Privacy Preserving Data Mining, vol. 19. Springer, New York (2006)
33. Zuech, R., Khoshgoftaar, T.M., Wald, R.: Intrusion detection and big heterogeneous data: a survey. J. Big Data **2**(1), 1–41 (2015)

Fine Grained Attribute Based Access Control Model for Privacy Protection

Que Nguyet Tran Thi(✉), Tran The Si, and Tran Khanh Dang

Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
{Ttqnguyet, khanh}@cse.hcmut.edu.vn,
513073035@hcmut.edu.vn

Abstract. Due to the rapid development of large scale and big data systems, attribute-based access control model has inaugurated a new wave in the research field of access control models. In this paper, we propose a novel and comprehensive framework for enforcing attribute-based security policies stored in JSON documents. We build a lightweight grammar for conditional expressions that are the combination of subject, resource, and environment attributes so that the policies are flexible, dynamic and fine grained. Moreover, with the approach of attribute-based access control, it can be applied to specify access purposes and intended purposes in purpose-based access control models for privacy protection. The experiment is carried out to illustrate the relationship between the processing time for access decision and the complexity of policies.

Keywords: Attribute based access control model · Purpose based access control model · Privacy protection · Privacy preserving

1 Introduction

Since the rapid development of large scale, open and dynamic systems, the shortcomings of traditional access control models (e.g. Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role based Access Control (RBAC) [1]) have gradually revealed, for example, applied for only closed systems, role explosion, complexity in compulsory assignments between users, roles, and permissions, and inflexibility in specifying dynamic policies and contextual conditions. Attribute based access control models (ABAC) have been recently investigated [2–4] and considered as one of three mandatory features for future access control systems [5].

Extensible Access Control Markup Language (XACML) 3.0 is an industrial OASIS standard[1] for enforcing access control policies based on attributes, considered as a predecessor of ABAC. In XACML policies, every operation on attributes even trivial conditions such as comparison requires function and data type definitions. This has caused the verbosity and difficulty in the specification of policies. Moreover, XACML is based on XML, which is not well-suited for Web 2.0 applications.

[1] Tran Khanh Dang: https://www.oasis-open.org/committees/xacml/.
Part of this work has been done on the secondment to Can Tho University of Technology.

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 305–316, 2016.
DOI: 10.1007/978-3-319-48057-2_21

In this paper, our access control model is built on the principle of NIST Standard ABAC that an access decision is *permitted* only if the request satisfies conditions on attributes of subject, resource and environment specified in policies. We also propose a light-weight grammar based on ANTLR for conditional expressions, which are human readable text and enough robust to describe complex policies such as user, data, environment driven policies. Besides, we extend ABAC for data privacy protection.

Privacy is a major concern in both of research and industrial fields due to dissemination of personal and sensitive data without user control, especially in mobile and ubiquitous computing applications and systems. In [6], privacy is defined as the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others. Most previous studies have considered privacy protection in access control models as constraints on purpose of data usage. This research area has recently drawn many interests, although it has developed since 2000 s. However, to the best of our knowledge, no research has integrated purpose-based access control into attribute-based access control model.

The rest of the paper is organized as follows. Section 2 gives a brief survey of related works. Section 3 presents the proposed model and introduces the main components. Section 4 describes the structure of policies. Section 5 indicates the mechanism of the proposed access control model in details. The experiment for evaluating the processing time is shown in Sect. 6. Concluding remarks and future work are discussed in Sect. 7.

2 Related Work

The development of Information Technology, especially in the age of Big Data and the Internet of Things environment, causes the role explosion problem and increases the complexity in permission management in the RBAC models which have been dominant for a long time [19, 20]. An emerging interest in addressing these problems is ABAC models, which can be adaptable with large, open and dynamic environments [2, 3]. In the common approach of ABAC, according to NIST standard [2], authorization decision is based on rules that simultaneously specify a set of conditions on numerous attributes such as subject, object, action and environment for a certain valid permission. Recent developments mostly focus on modeling attribute based access control to cover traditional models (i.e., DAC, MAC, RBAC) [4, 21–23]. Our paper takes a new approach at processing conditional expressions in attribute based policies and proposes an access control model integrating attribute based access control model with purpose based access control model.

Basically, a purpose compliance check in purpose based access control models depends on the relationship between access purposes and intended purposes of data objects ranging from the level of tables to the data cells [7–10]. In the beginning, Byun et al. [7] proposed the model with two types of allowable and prohibited intended purposes. It was then extended with an additional purpose, i.e. conditional intended purpose [9]. Several works have been conducted on enhancing this model by combining with role based access control (e.g., [11–14]), implementing with relational database management systems (DBMSs) with the technique of SQL query rewriting [15] and integrating with MongoDB [16]. Recently, action-aware with indirect access and direct access has also been considered in policies [17]. Nevertheless, the research works have not adopted purpose based access control model as attribute based access control model. Moreover, the declaration of purposes is mentioned mostly in relational database systems.

In summary, our work contributes a novel and comprehensive attribute based access control model with the new approach for policy specification. The proposed model also integrates the purpose concept to describe attribute based privacy policies for data privacy protection. Moreover, we use JavaScript Object Notation (JSON), which is a more light-weight data interchange format than XML and widely used in Web 2.0 applications as the language for policy specification.

3 FAPAC Components

In this section, we describe the proposed access control model. When a subject ***s*** accesses an object ***o***, the authorization process is carried out through two stages called as ***2-stage authorization***: (1) *access policy authorization* and (2) *privacy policy authorization*. The first step using access policies verifies that the request is legitimate with rights for the subject to access data. After that, the request is transferred to the second stage for checking privacy compliance based on privacy policies. The mechanism for authorization is described in details in Sect. 5.

The following part introduces the components shown in Fig. 1.

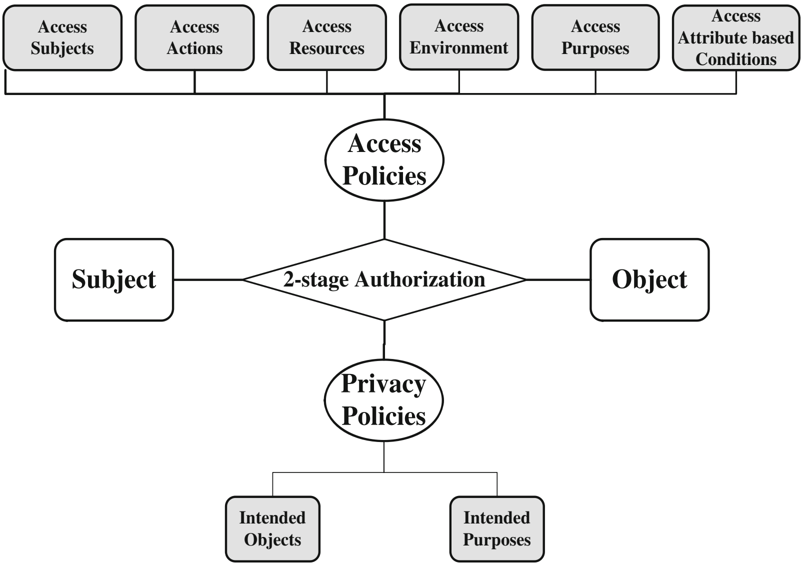


Fig. 1. The components of attribute based access control model for privacy protection

Access policies describe access permissions of subjects on resources, a set of permitted access purposes, and conditions based on attributes of subject, resource, and environment as well as obligations that are instructions from the Policy Decision Point module to the Policy Enforcement Point module to implement before or after an access is evaluated.

Privacy policies describe access restrictions on data objects which need to be preserved privacy. Privacy policies are specified through the assignment of intended purposes for data objects.

Access subjects are entities assigned rights to perform actions on objects.

Access conditions are Boolean combinations of subject, resource and environment attributes. The condition is specified according to a proposed grammar and parsed by using open source ANTLR[2].

Access Purposes and Intended Purposes. *Purpose* plays an important role in privacy policies to describe the reason for data usage. Purpose is classified into two types: *Access Purposes* and *Intended Purposes*. When users send a request to query data, they must provide their access purpose to the system. The access purpose is then verified whether the access subject is permitted for using it in the access policy authorization stage. Meanwhile, *Intended Purposes* is the set of allowable purposes of usage defined for data objects. Only data objects which intended purposes contain access purpose are valid for the request. The concept of purpose is borrowed from other research works in the series of purpose access control models [7, 8].

4 Policy Structure

In this section, we describe a general structure of attribute based policies and a detailed structure of access and privacy policies in the proposed model. The grammar for specifying conditional expressions in policies is also described in this section.

4.1 General Structure

In our system, a policy set includes policies. Each policy includes rules. Each rule defines a conditional expression that is a critical component in the policy. The rule returns a value specified in *Effect* if the condition is true. The target component, including three sub components *Subject*, *Action* and *Resource*, is used to select applicable policies for access decision. To avoid conflicts between policies and rules, the combining rule algorithms such as *permit override*, *deny override*, etc. are applied into the policy set and policies. The solution for using combining rule algorithm is inherited from XACML [26]. The relationship diagram between policy set, policies and rules are illustrated in the Fig. 2. The structure of a policy set is indicated by JavaScript Object Notation (JSON) as follows:

[2] www.antlr.org.

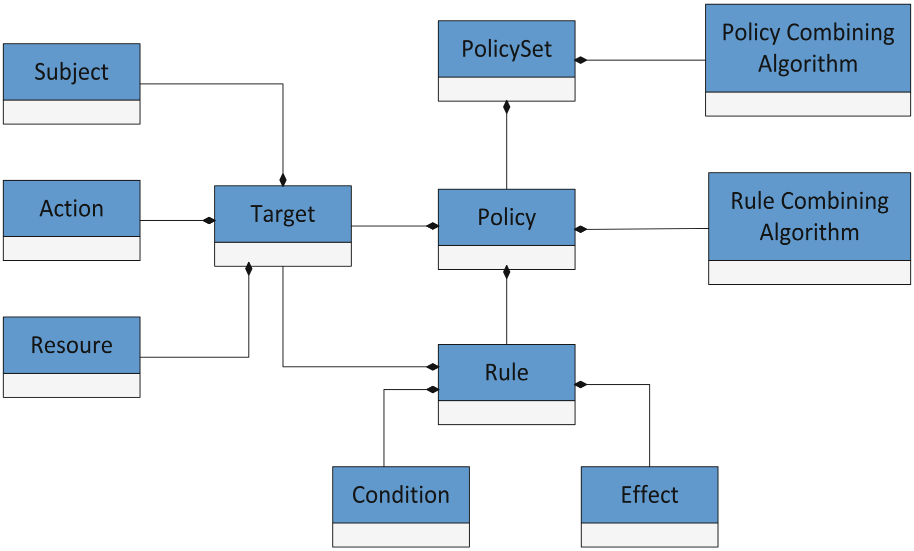
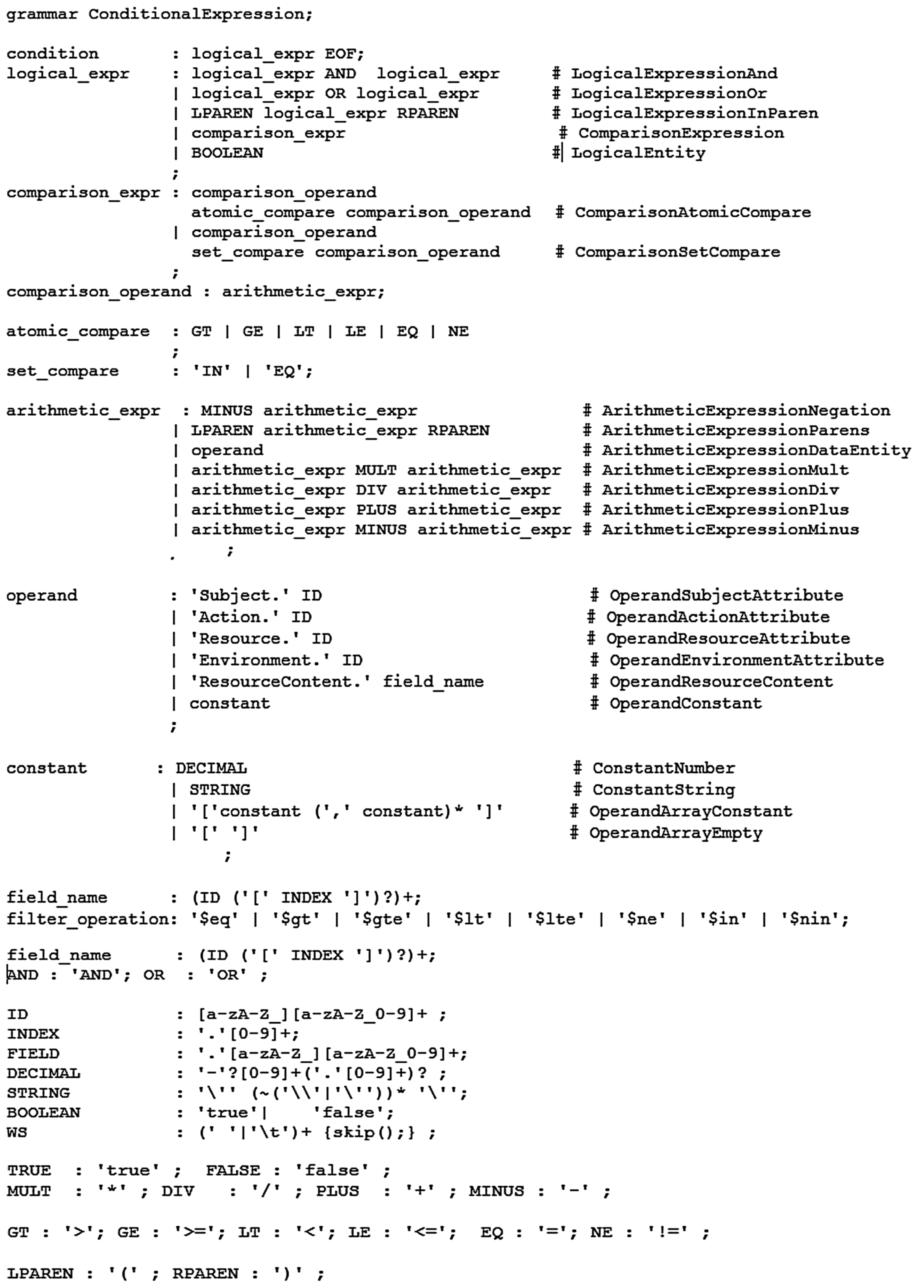


Fig. 2. The relationship diagram of components in policy set

```
CombiningRuleAlgID: <CombineAlgID>,
Policies: [{
      Policy: <policyID>,
      CombiningRuleAlgID: <CombineAlgID>,
       Target: {
          Subject: <subjectID>,
          Action: <actionID>,
          Resource: <resourceName>},
      Rules: [{
        RuleID: <ruleID>,
        Target: {
          Subject: <subjectID>,
          Action: <actionID>,
          Resource: <resourceName>},
        Condition: <conditionalExpr>,
        Effect: <effectValue>} ]
  }]}
```

In the above structure, the *condition* component < *conditionalExpr* > is written according to the below grammar:



```
grammar ConditionalExpression;

condition       : logical_expr EOF;
logical_expr    : logical_expr AND  logical_expr     # LogicalExpressionAnd
                | logical_expr OR logical_expr       # LogicalExpressionOr
                | LPAREN logical_expr RPAREN         # LogicalExpressionInParen
                | comparison_expr                     # ComparisonExpression
                | BOOLEAN                            #| LogicalEntity
                ;
comparison_expr : comparison_operand
                  atomic_compare comparison_operand   # ComparisonAtomicCompare
                | comparison_operand
                  set_compare comparison_operand      # ComparisonSetCompare
                ;
comparison_operand : arithmetic_expr;

atomic_compare  : GT | GE | LT | LE | EQ | NE
                ;
set_compare     : 'IN' | 'EQ';

arithmetic_expr  : MINUS arithmetic_expr                # ArithmeticExpressionNegation
                | LPAREN arithmetic_expr RPAREN         # ArithmeticExpressionParens
                | operand                               # ArithmeticExpressionDataEntity
                | arithmetic_expr MULT arithmetic_expr  # ArithmeticExpressionMult
                | arithmetic_expr DIV arithmetic_expr   # ArithmeticExpressionDiv
                | arithmetic_expr PLUS arithmetic_expr  # ArithmeticExpressionPlus
                | arithmetic_expr MINUS arithmetic_expr # ArithmeticExpressionMinus
                .     ;

operand         : 'Subject.' ID                          # OperandSubjectAttribute
                | 'Action.' ID                          # OperandActionAttribute
                | 'Resource.' ID                        # OperandResourceAttribute
                | 'Environment.' ID                      # OperandEnvironmentAttribute
                | 'ResourceContent.' field_name          # OperandResourceContent
                | constant                               # OperandConstant
                ;

constant       : DECIMAL                                # ConstantNumber
                | STRING                                # ConstantString
                | '['constant (',' constant)* ']'      # OperandArrayConstant
                | '[' ']'                               # OperandArrayEmpty
                     ;

field_name      : (ID ('[' INDEX ']')?)+;
filter_operation: '$eq' | '$gt' | '$gte' | '$lt' | '$lte' | '$ne' | '$in' | '$nin';

field_name       : (ID ('[' INDEX ']')?)+;
AND : 'AND'; OR  : 'OR' ;

ID               : [a-zA-Z_][a-zA-Z_0-9]+ ;
INDEX            : '.'[0-9]+;
FIELD            : '.'[a-zA-Z_][a-zA-Z_0-9]+;
DECIMAL          : '-'?[0-9]+('.'[0-9]+)? ;
STRING           : '\'' (~('\\'|'\''))* '\'';
BOOLEAN          : 'true'|    'false';
WS               : (' '|'\t')+ {skip();} ;

TRUE  : 'true' ;  FALSE : 'false' ;
MULT  : '*' ; DIV   : '/' ; PLUS  : '+' ; MINUS : '-' ;

GT : '>'; GE : '>='; LT : '<'; LE : '<=';  EQ : '='; NE : '!=' ;

LPAREN : '(' ; RPAREN : ')' ;
```

It can be seen that the operands in the condition expression are attributes from *Subject*, *Action*, *Resource*, *Resource Content* and *Environment* or specific values. The values of attributes are loaded from the request context. For missing values, the Policy Information Point module will look up from database to fulfill the request context. More details will be presented in Sect. 5.

An example for a rule is shown to demonstrate the above grammar:

```
//Description: Alice can read patient records only if she
is the doctor of patients.
RuleID: "ru001",
Target: {
    Subject: Alice,
    Action: read,
    Resource: patients},
Condition: Subject.role = "doctor" AND
           ResourceContent.doctorID = "Alice",
Effect: permit
```

4.2 Policies for Privacy Protection

To specify policies for privacy protection, *access purpose* is modeled as an attribute of environment and *intended purpose* is considered as an attribute of data objects in the resource content.

For example, the below policy indicates Alice has the privilege to access patient records with the purpose of research.

```
RuleID: "ru002",
Target: {
    Subject: Alice,
    Action: read,
    Resource: patients},
Condition: Environment.AccessPurpose = "research",
Effect: permit
```

Take an example that Alice sends a request to database to read patient records with the access purpose “research”. Then, the request context generated by the Policy Enforcement Point module is illustrated as follows:

```
Subject: {
  ID: "Alice",
  role: "researcher"},
Action: {
  ID: "read"},
Resource: {
  name: "patients",
  resourceContent: [
        {patientID: "001",
         patientName: "Bob",
         doctorID: "Alice",
         intendedPurpose: ["treatment", "research"]
       },
         {patientID: "002",
          patientName: "John",
          doctorID: "Alice",
          intendedPurpose: [ "treatment"]
        }]
},
Environment: {
  accessPurpose: "research"}
```

With the above request, the rule "ru002" returns permit and only the information of Bob is returned to Alice due to (1) Alice has the privilege of using the research access purpose, (2) the intended purposes of Bob contain "research". The detail of access control mechanism is described in Sect. 5.

5 Attribute Based Access Control Mechanism for Privacy Protection

In this work, we utilize the concepts in XACML such as Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Information Point (PIP), Policy Administration Point (PAP) and Obligations. Due to space limitation, we do not explain the functions of these components. More details can be seen at [26]. The below section will describe the data flow of our model.

The main processes in the data flow of our model depicted in Fig. 3 are described as follows:

1. **PEP** receives the access request consisting of the components: *subject, action*, and *resource*.
2. **PEP** creates another request, called *request context,* for policy decision from the access request fulfilled with the attributes of subject, action, resource, and environment and then sends to **PDP** for access authorization.
3. **PDP** retrieves the list of access policies from database.

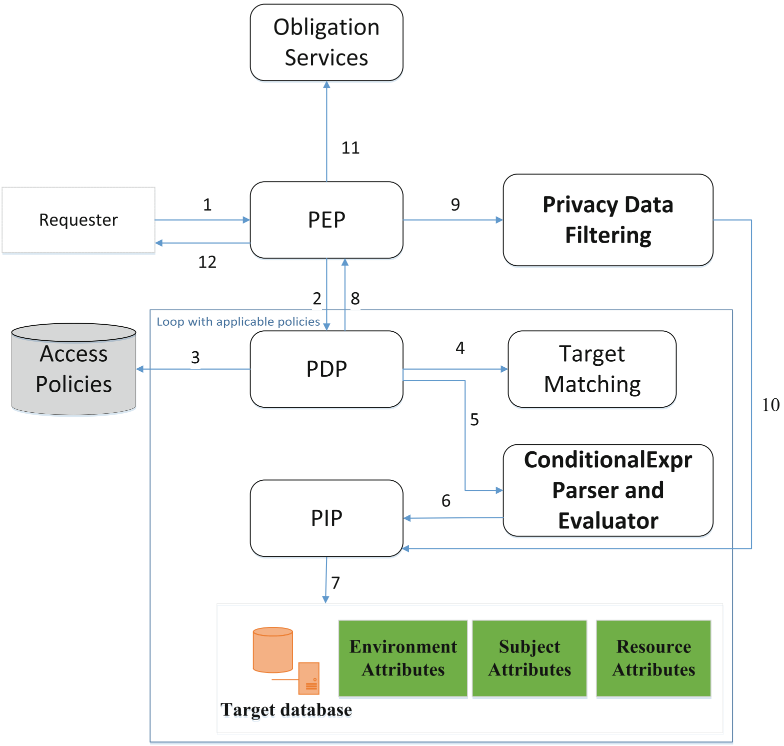


Fig. 3. The data flow diagram of the proposed access control model

4. For each policy, **PDP** checks whether the *target* element of the policy (i.e. *subject, action, resource*) matches with the corresponding components of the request context by the *Target Matching* module. If it returns "*successfully matching*", all rules of this policy are examined. Rules satisfying the *target* component will be processed in the next step.
5. In this step, the *condition* component of applicable rules is evaluated by the module **ConditionalExpr Parser and Evaluator**. **ConditionalExpr Parser and Evaluator** uses the open source *ANTLR* with the grammar presented in Sect. 4.1 to evaluate the expression.

Depending on the combining rule algorithm specified in the component CombiningRuleAlgID in the current policy, PDP will continue to check the next access rule (e.g. permit overrides) or terminate with the result deny (e.g. deny overrides). Similarly, depending on the combining algorithm for policies specified in the component CombiningRuleAlgID of PolicySet, PDP will stop with the result of policy evaluation or keep on checking with other applicable policies.

6. The module **Condition Parser and Evaluation** sends requests to *Policy Information Point* (**PIP**) to retrieve values for operands.

7. **PIP** collects values from the request context and database; and then sends them to **ConditionalExpr Parser and Evaluator** for expression evaluation.
8. After evaluating all applicable policies from step 4 to step 7, **PDP** returns the response to **PEP**. The response can be *permit* or *deny*.
9. In the case of *permit*, **PEP** asks the module **Privacy Data Filtering** for filtering data in *ResourceContent* which are valid for the access purpose.
10. In the case that *ResourceContent* has not been fulfill in the request context due to no policy requires resource data, the **Privacy Data Filtering** module sends the request to PIP for querying resource content.
11. Finally, **PEP** calls **Obligation Services** to perform obligations. Depending on specification, the obligations are executed before or after **PEP** returns results to the requester. The details can be referred at [26].
12. **PEP** returns data results to the requester.

In this section, we presented the data flow for processing policies in the proposed attribute based access control model for privacy protection. A prototype with the basic modules (e.g. PDP and PIP) for access decision is also implemented. The next section describes several results for evaluating the solution.

6 Evaluation

We carried out experiments about the relevance between processing time of the PDP module and complexity of rules. The system configuration for the experiments is Dell Vostro 3650, 8 GB RAM, Intel core i5-3230 M 2.60 GHz. The prototype is implemented by Java SDK, Spring Framework and MongoDB 3.0.7 for storing policies and data. The target database includes 20 collections with 20 attributes and 200 documents for each collection generated randomly. Each subject, action, resource and environment contains 20 attributes.

The following table indicates the results after six experiments. For each experiment, we measure processing time with three times (e.g. T1, T2 and T3). From the Table 1, it can be seen that the processing time increases with the complexity of rules. However, when there was a fivefold and twofold increase in the number of rules from 10 to 50 and from 50 to 100, the processing time only increased by 2–3 ms, approximately 13–25 %.

Table 1. The results of experiments

ID	Number of rules	Logical expressions in each rule	Arithmetic expressions in each rule	T1 (ms)	T2 (ms)	T3 (ms)
1	1	1	1	5	4	4
2	1	5	5	6	6	6
3	1	10	10	8	8	7
4	10	10	10	12	11	12
5	50	10	10	15	14	14
6	100	10	10	17	17	16

About the fine granularity, our attribute based access control model can describe various attribute based policies due to the flexibility of conditional expressions built under the proposed grammar. Compared to other approaches, our model can specify policies with the conditions based on the combination of attributes of subject, resource, and environment. Besides, by using JSON, our model can easily integrate with Web 2.0 applications as web as devices and systems in Internet of Things.

7 Conclusion

In this paper, we have proposed the fine grained attribute and purpose based access control model for privacy protection with the mechanism of 2-stage authorization. A conditional expression based on attributes of subject, resource, action and environment are built on the ANTLR grammar, which is enough to describe various policies. Modeling access purpose as the attribute of environment and intended purposes as the attribute of data objects in the resource makes the mechanism simpler but effective and consistent. In future, we will improve the grammar for the conditional expression to describe more complex policies which can retrieve data from multiple sources not only from the request context. Besides that, intended purposes will be extended for data cells to increase the fine grained level of the proposed access control model.

References

1. Bertino, E., Ghinita, G., Kamra, A.: Access Control for Databases: Concepts and Systems. Now Publishers, Hanover (2011)
2. Hu, V.C., Ferraiolo, D., Kuhn, R., Friedman, A.R., Lang, A.J., Cogdell, M.M., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K.: Guide to Attribute Based Access Control (ABAC) definition and considerations (draft). NIST Spec. Publ. **800**, 162 (2013)
3. Hu, V.C., Kuhn, D.R., Ferraiolo, D.F.: Attribute-based access control. Computer **2**, 85–88 (2015)
4. Jin, X., Krishnan, R., Sandhu, R.: A unified attribute-based access control model covering DAC, MAC and RBAC. In: Cuppens-Boulahia, N., Cuppens, F., Garcia-Alfaro, J. (eds.) DBSec 2012. LNCS, vol. 7371, pp. 41–55. Springer, Heidelberg (2012). doi:10.1007/978-3-642-31540-4_4
5. Sandhu, R.: The future of access control: attributes, automation, and adaptation. In: Sai Sundara Krishnan, G., Anitha Lekshmi, R.S., Senthil Kumar, M., Bonato, A., Graña, M. (eds.) Computational Intelligence, Cyber Security and Computational Models, vol. 246, p. 45. Springer, India (2013)
6. Westin, A.F.: Privacy and Freedom. Atheneum, New York (1967)
7. Byun, J.-W., Bertino, E., Li, N.: Purpose based access control of complex data for privacy protection. In: Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies (2005)
8. Byun, J.W., Li, N.: Purpose based access control for privacy protection in relational database systems. VLDB J. **17**(4), 603–619 (2008)

9. Kabir, M.E., Wang, H.: Conditional purpose based access control model for privacy protection. In: Proceedings of the Twentieth Australasian Conference on Australasian Database, vol. 92, pp. 135–142. Australian Computer Society, Inc. (2009)
10. Wang, H., Sun, L., Bertino, E.: Building access control policy model for privacy preserving and testing policy conflicting problems. J. Comput. Syst. Sci. **80**(8), 1493–1503 (2014)
11. Kabir, M.E., Wang, H., Bertino, E.: A role-involved conditional purpose-based access control model. In: Janssen, M., Lamersdorf, W., Pries-Heje, J., Rosemann, M. (eds.) E-Government, E-Services and Global Processes, vol. 334, pp. 167–180. Springer, Heidelberg (2010)
12. Kabir, M.E., Wang, H., Bertino, E.: A conditional purpose-based access control model with dynamic roles. Expert Syst. Appl. **38**(3), 1482–1489 (2011)
13. Ni, Q., Lin, D., Bertino, E., Lobo, J.: Conditional privacy-aware role based access control. In: Biskup, J., López, J. (eds.) ESORICS 2007. LNCS, vol. 4734, pp. 72–89. Springer, Heidelberg (2007). doi:10.1007/978-3-540-74835-9_6
14. Ni, Q., Bertino, E., Lobo, J., Brodie, C., Karat, C.M., Karat, J., Trombeta, A.: Privacy-aware role-based access control. ACM Trans. Inf. Syst. Secur. (TISSEC) **13**(3), 24 (2010)
15. Colombo, P., Ferrari, E.: Enforcement of purpose based access control within relational database management systems. IEEE Trans. Knowl. Data Eng. **26**(11), 2703–2716 (2014)
16. Colombo, P., Ferrari, E.: Enhancing MongoDB with purpose based access control. IEEE Trans. Dependable Secure Comput. (2015, to appear)
17. Colombo, P., Ferrari, E.: Efficient enforcement of action-aware purpose-based access control within relational database management systems. IEEE Trans. Knowl. Data Eng. **27**(8), 2134–2147 (2015)
18. Pervaiz, Z., Aref, W.G., Ghafoor, A., Prabhu, N.: Accuracy-constrained privacy-preserving access control mechanism for relational data. IEEE Trans. Knowl. Data Eng. **26**(4), 795–807 (2014)
19. Ferraiolo, D.F., Sandhu, R., Gavrila, S., Kuhn, D.R., Chandramouli, R.: Proposed NIST standard for role-based access control. ACM Trans. Inf. Syst. Secur. (TISSEC) **4**(3), 224–274 (2001)
20. Fuchs, L., Pernul, G., Sandhu, R.: Roles in information security–a survey and classification of the research area. Comput. Secur. **30**(8), 748–769 (2011)
21. Kuhn, D.R., Coyne, E.J., Weil, T.R.: Adding attributes to role-based access control. IEEE Comput. **43**(6), 79–81 (2010)
22. Huang, J., Nicol, D.M., Bobba, R., Huh, J.H.: A framework integrating attribute-based policies into role-based access control. In: Proceedings of the 17th ACM Symposium on Access Control Models and Technologies, pp. 187–196. ACM (2012)
23. Rajpoot, Q.M., Jensen, C.D., Krishnan, R.: Attributes enhanced role-based access control model. In: Fischer-Hübner, S., Lambrinoudakis, C., López, J. (eds.) TrustBus 2015. LNCS, vol. 9264, pp. 3–17. Springer, Heidelberg (2015)
24. Sweeney, L.: Achieving K-anonymity privacy protection using generalization and suppression. Int. J. Uncertainty Fuzziness Knowl. Based Syst. **10**(5), 571–588 (2002)
25. Ni, Q., Bertino, E., Lobo, J.: An obligation model bridging access control policies and privacy policies. In: Proceedings of the 13th ACM Symposium on Access Control Models and Technologies, pp. 133–142 (2008)
26. Rissanen, E.: eXtensible Access Control Markup Language (XACML) version 3.0 (committe specification 01). Technical report, OASIS (2010). http://docs.oasisopen.org/xacml/3.0/xacml-3.0-core-spec-cd-03-en.Pdf

A Supporting Automatically Mechanism for Data Owner Preventing Personal Privacy from Colluding Attack on Online Social Networks

Nguyen Hoang Nam Pham[1(✉)], Thanh Tien Nguyen[2],
and Thi Kim Tuyen Le[3]

[1] Department of Information Technology,
University of Economic Ho Chi Minh City, Ho Chi Minh City, Vietnam
nam@ueh.edu.vn
[2] IT and Data Management Department,
Oxford University Clinical Research Unit, Ho Chi Minh City, Vietnam
tiennt@oucru.org
[3] Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh City, Vietnam
tuyenltk@cse.hcmut.edu.vn

Abstract. Worldwide, there are over 1.65 billion monthly active Facebook users (MAUs) which is a 15 % increase year over year. What this means for you: in case you meet any lingering doubts, Facebook is too big to be ignored. Online social networks (OSNs) offer a useful environment for our social purposes such as sharing information and communicating to with each other. However, OSNs suffer also high risk of leakage private user information. In this paper, we present a mechanism for data owner preventing automatically personal privacy from colluding attack. We approach supporting automatically making approval for new relationship to shared data basing on historical data.

Keywords: Access control · Online social network · Colluding attack

1 Introduction

In recent years, we have seen a surprising growth of OSNs, and OSNs become more popular on internet. [1] Following Facebook's report in 27 April 2016, worldwide, there are over 1.65 billion monthly active Facebook users (MAUs) which is a 15 % increase year over year. 1.09 billion users log onto Facebook daily (DAU) for March 2016, which represents a 16 % increase year over year. The Implication: A huge and vastly growing number of Facebook users are active and consistent in their visits to the site, making them a promising audience for your marketing efforts. Every 60 s on Facebook: 510 comments are posted, 293,000 statuses are updated, and 136,000 photos are uploaded. Again, there are many active users, but also a huge amount of information competing for their attention, so quality and strategy on your part matter. 4.75 billion pieces of content are shared daily as of May 2013 which is a 94 % increase from August 2012.

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 317–328, 2016.
DOI: 10.1007/978-3-319-48057-2_22

This means we have a huge amount of privacy information including profile data, location data, and activities on OSNs. It is a major difficulty to control sharing data to each other with a trade-off between privacy risk and social benefit. Current OSNs provide us a mechanism for allowing or denying what friends can access what data. However, users on OSNs often do not care who can access their shared information. Further, considered users may take mistakes in sharing data because of lacking of security skills or no concern about the importance of sharing.

On OSNs, a piece of content belongs to particular people having various connections. For example, Alice posts an image which Bob is tagged in; Carol likes a Dave's comment about some articles else. Thus, concerned users expect to have a mechanism to define their own access control policies. Users regulate access control policies for the same shared data, so policy conflict is inevitable. Therefore, OSNs have trend to put considered policies together to harmonize users' social purpose with privacy information of each particular people. Malicious users make use of collaboration data sharing on OSNs for colluding to each other to disclose privacy information.

The remainder of this paper is organized as follows: Sect. 2 overviews online social networks and access control models for them. In Sect. 3, we discuss and present our mechanism Sect. 4 presents the experimental results. Finally, we discuss conclusion and future work in Sect. 5.

2 Related Works

2.1 Online Social Networks

Boyd and Ellison [2] defined social network sites (also described as "online social network" or "social networking services") as web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system. OSNs are graph data structures. They consist of a finite (and possibly mutable) set of vertices (presenting the list of users and the list of resource), together with a set of ordered pairs of these vertices known as edges (presenting the three types of relationship on OSNs: user-to-user, resource-to-resource, and user-to-resource).

OSNs always encourage users give information as more as possible by declaring user profile, making new connections. The more frequently you have activity on OSNs, the more valuable information you supply in OSNs. Analyzers will analyze this valuable information on OSNs to find out interesting user trends. E.g. Which subjects are Vietnamese youth interested in? Shopping trends and predictions of U.S. So, it makes a fast developing of social services for particular purpose (entertainment, travel, shopping, food & drink, dating, etc.). E.g. If you like travel a lot for going somewhere to discover lifestyle, and interesting places, a travel social networks will be an excellent tools for you to browser some recommended information, choose future target, review hotels, public transportation, etc. However, for this travel OSN is more convenient and useful, you have to give some information to this OSN, such as: user profile, photos, historical search places, making review, or rating on historical places which you went to.

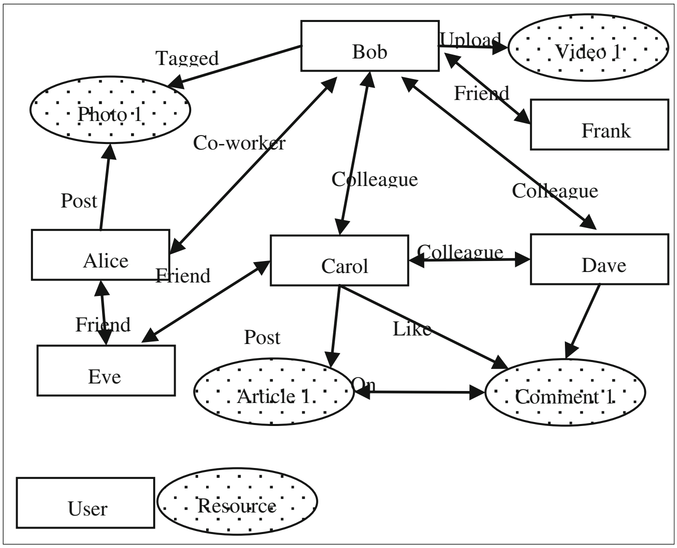


Fig. 1. A sample of online social networks

Thus, other can know your interest, your current location, etc. On the other hand, users have a high risk of disclosing personal information (Fig. 1).

Protecting privacy problem on OSNs was interesting, and attracted attention to research community soon when it was presented in 1992. Several models have been proposed to address collaborative systems, usually focusing more on groupware-like applications. Notable models include Task-based access control [3, 4] and Team-based access control [5]. Those models are more appropriate for social networks, they are still lacking in several respects. Simple ACMs have the advantage of being straightforward; on the other hand, they are not flexible enough. To address these questions, some classical ACMs have been incorporated in SNs with specific aspects added to the underlying mechanisms. Other approaches emerged to fit different requirements of OSNs.

Ralph Gross et al. [6] presented some researches about revelation personal information and privacy on OSNs with case study – Facebook. In this work, they raised the problems which have privacy implications, such as: stalking when using an instant messenger application, data re-identification, and some fragile privacy protection problems. Then, Yan Li et al. [7] analyzed them same problem more detail, and described it systematically. They classified personal information on OSNs in to 3 types: Personal particular (current city, hometown, sex, birthday, relationship status, high school, music, books, email, etc.), Social Relationship (incoming friend list, outgoing friend list), Social Activities (status message, photo, link, comments, like, etc.). Users on OSNs share their information, interaction with each other by 4 basic functionalities: Publish, Recommend, Tag and Push. E.g. Alice publishes her personal information; Bob's social relationships are recommended to Alice, Alice tags Bob in her social activity, Bob's personal information is pushed to Alice's feed page when Bob publishes his personal information. They have analyzed the exploits and attacks which may lead to privacy leakage and to find out suggestions on mitigating the corresponding exploits and attacks. They are inferable personal particular, cross-site incompatibility, inferable social relationship, unregulated relationship recommendation, inferable social activity, ineffective rule update, invalid hiding list.

2.2 Access Control for Online Social Networks

As we discussed, protecting privacy problem on OSNs was interesting for research community. There are a lots access control model with various approaches were proposed for OSNs. Lorena González-Manzano [8] classified those models into 5 groups:

- Role-Based Access Control: According to Jianguo Li et al. [9], the user plays a social role when s/he enters a social network site or creates relationships with other users. For example, when a user enters a university, he gets a social role "student". His relationship with other user which has "student" role is "classmate" or "alumnus", and the relationship with other user which has "teacher" role is "teacher-student". Thus roles have a relational nature and imply patterns of relationships. Users who are closely related with each other or belong to the same organization will form a social community (known a group). Every group has a particular role which associates the group members. When the user enters the group, he acquires the Group Role.
- Trust-Based Access Control, Barbara Carminati et al. [10] present a model, where policies are expressed as constraints on type, depth, and trust level of existing relationships. This model allows the specification of access rules for online resources which are expressed in Notation 3 Logic (N3), and then evaluated by Cwm reasoner against the existing relationships in order to generate a proof.Jennifer Ann Golbeck [11] presents computing and applying trust in web-based social networks. In her dissertation, she apply approach into some specific domain, such as: "TrustFilm" this is an application, This is an website that combines trust networks with movie information (user's movie rating, and their friends'). This OSNs will collect and analysis trust data to recommend concerned film to someone.
- Attribute-Based Access Control
- Relationship-Based Access Control
- And, Ontology-Based Access Control Carminati et al. [12] propose a rule-based model that defines authorization, administration and filtering policies based on trust relationships. The model encodes user profiles, relationships, objects and actions in an ontology-based hierarchy. The model offers a high degree of flexibility in terms of defining hierarchies of user types, relations, objects and actions as well as adding new ontology to the knowledge base. The hierarchical structure enables flexible rules propagation on concepts. OSNAC, an Ontology-Based Access Control Model for Social Networking Systems [13], is more detailed than the previously discussed model [14]. Masoumzadeh et al. focus in this model on relationship protection. For this purpose, the concept annotation is incorporated in access rules, which is a type of a digital object representing a relationship between an object and a subject.

Access rules define negative or positive authorization, which can be simple or advanced when composed by means of disjunctions or conjunctions yielding multiple authorization rules and delegation of authority from one user to another.

The tendency of access control model for OSNs is incorporating models by utilizing advantages of those to meet current requirements of OSNs. E.g. Yuan Cheng et al. [15] extend their model by adding attribute factor to propose Attribute-aware Relationship-based Access. Control for Online Social Networks. Bruns et al. [16] fused reason rule factor into [17] to propose Relation based access control through hybrid logic.

2.3 Access Control for Collaboration Environment

Because each user may have various relationship to the same resource, so each user desires to regular particular access policies for the same resource. So, each user has a certain role in control how to access that resource. In that case, OSNs have features of collaboration environment. Hongxin Hu [18] proposed Multiparty Access Control for OSNs. In their model, they pre-defined 4 types of relationship between user-to-resource (Owner, Contributor, Stakeholder, and Disseminator). They describe the processing of their model into 2 phases: evaluation, decision aggregation (Fig. 2).

Whenever an access request occur, user firstly has one of 4 types of user-to-resource relationship (Owner, Contributor, Stakeholder and Disseminator) with resource, will evaluate himself/herself policies to make partial decision deny or allow access information. Secondly, system will aggregate all partial decision which has concern with to make final decision. Policy conflict is inevitable. They present some mechanisms can be apply to solve this problem, including a voting scheme for decision making of multiparty control, threshold-based conflict resolution, strategy-based conflict resolution with recommendation by calculating sensitivity score of each controller on the shared data. Finally, they have proof the correctness of their mechanism.

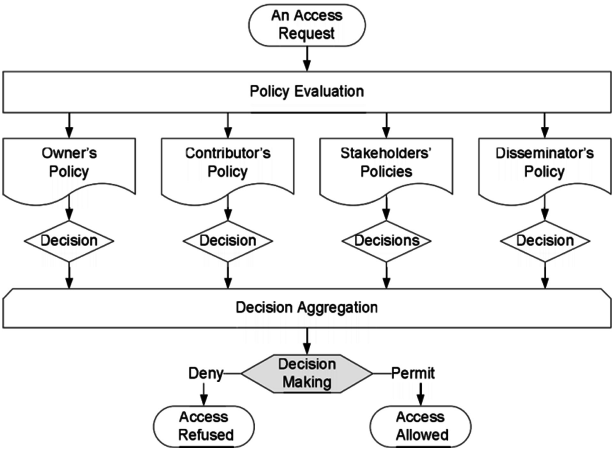


Fig. 2. Multiparty policy evaluation process

3 Our Approach

The key process of this mechanism is analyzing on the historical data between data owner, existing stakeholders and new candidate stakeholder, when this new candidate stakeholder requires making a new relationship to something. A candidate stakeholder is reasonable, if he has a reasonable count of historical data between them. Otherwise, he is in one of three situations:

– He is a stranger (a newcomer or a new friend), so he has too little relationship with others.
– He has less activity in OSNs, so that he has little common data with them.
– He is a malicious user who colludes with others stakeholder to disclose data owner's personal information.

Our mechanism will calculate automatically the similarity value of new candidate stakeholder whenever occurrence of request a new connection. The less similarity value is, the more similar candidate stakeholder is, and the more opportunity request will be approved.

Given:

- o is the data owner of data d,
- $S_= \{s_i \mid i \le n\}$ is the set of n stakeholders of data d,
- c is the candidate stakeholder who want to make a new connection to data d.

Our mechanism processes sequentially through 4 steps below:

Step 1: Select m concerned data item d, which satisfy two conditions:

– Its type is the same type of owner's shared data.
– Its minimum common relationships equals 2, because we cannot require make new relationships for the person who has only 1 common relationship to each other.

Step 2: Fill weighted value in below table (*n + 2* columns, *m + 1* rows) by these rules (Table 1):

$$w(j,i) = \begin{cases} 0, j > 0 \wedge relationship(s_i, d_j) \in R \\ 0.9, j > 0 \wedge relationship(s_i, d_j) \notin R \end{cases}$$

Step 3: Calculate similarity value of each item d_j with item d by Distance Formula:

$$similarity(d_a, d_b) = \sqrt{\sum_{n+1} (w(a,i) - w(b,i))^2}$$

This means the similarity of 2 data items. The more minimum similarity value is, the closer they are.

Table 1. Weight matrix summarizing relations between m data item

	o	s_1	**…**	s_i	**…**	s_n	***c***
d	0	0	…	0.9	…	0.9	0
d_1	0	0.9	…	0	…	0.9	0.9
…	…	…	…	…	…	…	…
dj	n	0	…	0	…	0	0
…	…	…	…	…	…	…	…
d_m	n	0	…	0	…	0.9	0

Step 4: Calculate some average similarity value

$$real\_AVG\_similarity = \frac{\sum_{j \to m} Similarity(d_j, d)}{m}$$

Some Special Cases:

- When each data item d_j has only relationship with candidate stakeholder c, and data owner o.

$$case\_1\_AVG\_similarity = \frac{\sum_{j \to m} Similarity(d_j, d)}{m} = \frac{\sum_{j \to m} \sqrt{\sum_{n} (0.9)^2}}{m} = 0.9\sqrt{n}$$

- When each data item d_j has relationship with candidate stakeholder c, and every stakeholders with the exception of data owner o. (the colluding case)

$$case\_2\_AVG\_similarity = \frac{\sum_{j \to m} Similarity(d_j, d)}{m} = \frac{\sum_{j \to m} \sqrt{n^2}}{m} = n$$

- When each data item d_j has relationship with candidate stakeholder c, and everyone (the most similar case)

$$case\_3\_AVG\_similarity = \frac{\sum_{j \to m} Similarity(d_j, d)}{m} = \frac{\sum_{j \to m} \sqrt{(n+2) * 0^2}}{m} = 0$$

- When each data item d_j has only relationship with candidate stakeholder c, and with one stakeholder s (the most different case)

$$case\_4\_AVG\_similarity = \frac{\sum_{j \to m} Similarity(d_j,d)}{m} = \frac{\sum_{j \to m} \sqrt{n^2 + (n-1)*0.9^2}}{m} = \sqrt{n^2 + (n-1) * 0.9^2}$$

- When each data item d_j has only relationship with candidate stakeholder c, and with data owner o, and with one stakeholder s.

$$case\_5\_AVG\_similarity = \frac{\sum_{j \to m} Similarity(d_j,d)}{m} = \frac{\sum_{j \to m} \sqrt{(n-1)*0.9^2}}{m} = \sqrt{(n-1) * 0.9^2} = 0.9\sqrt{(n-1)}$$

Step 5: Make decision on creating a new relationship, based on conditional formula:

$$\begin{cases} allow : real\_AVG\_similarity \le \frac{1}{3}\left(\sum_{i \in \{1,2,4\}} case\_i\_AVG\_similarity\right) \\ ?ask\_data\_owner : real\_AVG\_similarity > \frac{1}{3}\left(\sum_{i \in \{1,2,4\}} case\_i\_AVG\_similarity\right) \end{cases}$$

4 Experiments and Results

We are experimenting with the case (given k common stakeholders of data d is between 1 and 12; and n stakeholders, and m historical selected data item is 90) in order to examine and approve our mechanism (Tables 2, 3 and 4).

Finally, we choose a specific case (when *k* – the number of common stakeholders is 4, and the number of common historical data is from 7 to 90) to summarize our method. In this chart, value of *c* + *o* is always less than value of *c* + *k(s)*, because *c* + *o* case is more similar than *c* + *k(s)*. We find out avg value by averaging to examine on allowing to create a new relationship (Fig. 3).

Table 2. Experiments when (candidate stakeholder and data owner) historical selected data item is between 7 and 90; common users is between 1 and 12.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
7	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6
15	5.4	5.4	5.4	5.4	5.4	5.4	5.4	5.4	5.4	5.4	5.4	5.4
22	3.7	3.7	3.7	3.7	3.7	3.7	3.7	3.7	3.7	3.7	3.7	3.7
30	2.7	2.7	2.7	2.7	2.7	2.7	2.7	2.7	2.7	2.7	2.7	2.7
37	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2
45	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8
52	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6
60	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
67	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2
75	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1
82	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
90	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9

Table 3. Experiments when (candidate stakeholder and *k* other stakeholders) historical selected data item is between 7 and 90; common users is between 1 and 12.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
7	159.0	158.6	158.1	157.7	157.3	156.9	156.4	156.0	155.6	155.2	154.7	154.3
15	74.2	74.0	73.8	73.6	73.4	73.2	73.0	72.8	72.6	72.4	72.2	72.0
22	50.6	50.5	50.3	50.2	50.0	49.9	49.8	49.6	49.5	49.4	49.2	49.1
30	37.1	37.0	36.9	36.8	36.7	36.6	36.5	36.4	36.3	36.2	36.1	36.0
37	30.1	30.0	29.9	29.8	29.8	29.7	29.6	29.5	29.4	29.4	29.3	29.2
45	24.7	24.7	24.6	24.5	24.5	24.4	24.3	24.3	24.2	24.1	24.1	24.0
52	21.4	21.3	21.3	21.2	21.2	21.1	21.1	21.0	20.9	20.9	20.8	20.8
60	18.5	18.5	18.5	18.4	18.4	18.3	18.3	18.2	18.2	18.1	18.1	18.0
67	16.6	16.6	16.5	16.5	16.4	16.4	16.3	16.3	16.3	16.2	16.2	16.1
75	14.8	14.8	14.8	14.7	14.7	14.6	14.6	14.6	14.5	14.5	14.4	14.4
82	13.6	13.5	13.5	13.5	13.4	13.4	13.4	13.3	13.3	13.2	13.2	13.2
90	12.4	12.3	12.3	12.3	12.2	12.2	12.2	12.1	12.1	12.1	12.0	12.0

Table 4. Experiments when (candidate stakeholder, data owner, and *k* other stakeholders) historical selected data item is between 7 and 90; common users is between 1 and 12.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
7	38.4	36.6	34.7	32.7	30.6	28.3	25.9	23.1	20.0	16.4	11.6	0.0
15	17.9	17.1	16.2	15.3	14.3	13.2	12.1	10.8	9.4	7.6	5.4	0.0
22	12.2	11.6	11.0	10.4	9.7	9.0	8.2	7.4	6.4	5.2	3.7	0.0
30	9.0	8.5	8.1	7.6	7.1	6.6	6.0	5.4	4.7	3.8	2.7	0.0
37	7.3	6.9	6.6	6.2	5.8	5.4	4.9	4.4	3.8	3.1	2.2	0.0
45	6.0	5.7	5.4	5.1	4.8	4.4	4.0	3.6	3.1	2.5	1.8	0.0
52	5.2	4.9	4.7	4.4	4.1	3.8	3.5	3.1	2.7	2.2	1.6	0.0
60	4.5	4.3	4.1	3.8	3.6	3.3	3.0	2.7	2.3	1.9	1.4	0.0
67	4.0	3.8	3.6	3.4	3.2	3.0	2.7	2.4	2.1	1.7	1.2	0.0
75	3.6	3.4	3.2	3.1	2.9	2.6	2.4	2.2	1.9	1.5	1.1	0.0
82	3.3	3.1	3.0	2.8	2.6	2.4	2.2	2.0	1.7	1.4	1.0	0.0
90	3.0	2.8	2.7	2.5	2.4	2.2	2.0	1.8	1.6	1.3	0.9	0.0

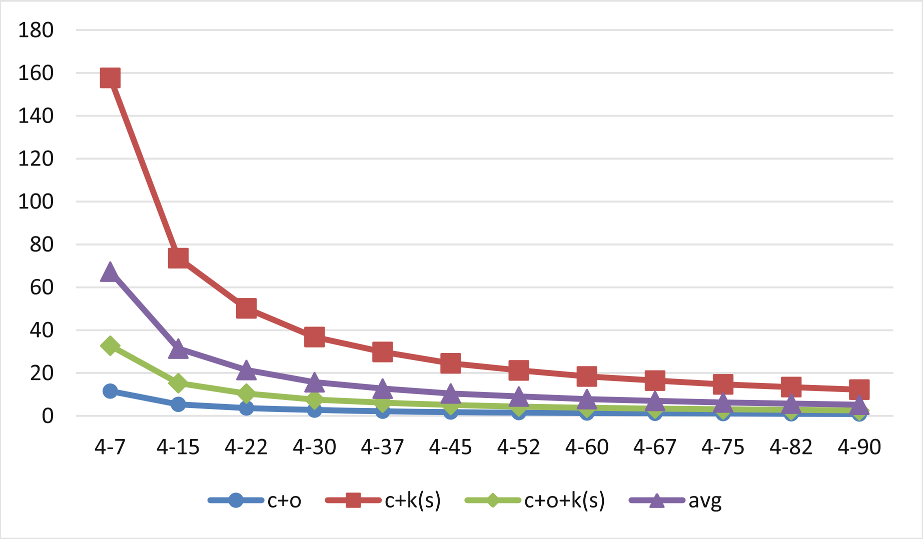


Fig. 3. Experiments (three cases and average value) when historical selected data item is between 7 and 90; real common users is 4 over 12.

5 Conclusion and Future Work

In this paper, we present a supporting automatically mechanism for data owner preventing personal privacy from colluding attack on Online Social Networks. This method can automatically approve or deny making relationship by calculating historical data. In our future work, we will study how to determine the threshold values used in this paper effectively and efficiently according to Social networks sample data from Stanford Large Network Dataset Collection [19].

References

1. https://zephoria.com/top-15-valuable-facebook-statistics/
2. Boyd, D.M., Ellison, N.B.: Social network sites: definition, history, and scholarship. J. Comput. Mediated Commun. **13**, 210–230 (2007)
3. Thomas, R.K., Sandhu, R.S.: Conceptual foundations for a model of task-based authorizations. In: 7th IEEE Computer Security Foundations Workshop, pp. 66–79. IEEE Computer Society Press (1994)
4. Thomas, R.K., Sandhu, R.S.: Task-based authorization controls (TBAC): a family of models for active and enterprise-oriented authorization management. In: Proceedings of the IFIP TC11 WG11.3 Eleventh International Conference on Database Security XI: Status and Prospects, pp. 166–181. Chapman & Hall, Ltd., London. ISBN 0-412-82090-0 (1998)
5. Thomas, R.K.: Team-based access control (TMAC): a primitive for applying role-based access controls in collaborative environments. In: Second ACM Workshop on Role-Based Access Control, pp. 13–19. ACM (1997). ISBN 0897919858
6. Gross, R., Acquisti, A.: Information revelation and privacy in online social networks. In: Proceedings of the 2005 ACM workshop on Privacy in the Electronic Society (WPES 2005), pp. 71–80 (2005). ISBN 1-59593-228-3
7. Li, Y., Li, Y., Yan, Q., Deng, R.H.: Privacy leakage analysis in online social networks. Comput. Secur. **49**, 239–254 (2015). http://dx.doi.org/10.1016/j.cose.2014.10.012
8. González-Manzano, L., González-Tablas, A.I., de Fuentes, J.M., Ribagorda, A.: SoNeUCONABC, an expressive usage control model for Web-Based Social Networks. Comput. Secur. **43**, 159–187 (2014). http://dx.doi.org/10.1016/j.cose.2014.03.009
9. Li, J., Tang, Y., Mao, C., Lai, H., Zhu, J.: Role based access control for social network sites. In: Joint Conferences on Pervasive Computing (JCPC) 2009, Tamsui, Taipei, pp. 389–394 (2009)
10. Carminati, B., Ferrari, E., Perego, A.: Rule-based access control for social networks. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006. LNCS, vol. 4278, pp. 1734–1744. Springer, Heidelberg (2006). doi:10.1007/11915072_80
11. Golbeck, J.: Computing and applying trust in web-based social network. Ph.D. thesis, University of Maryland, College Park, Md, USA (2005)
12. Fong, P.W.L.: Relationship-based access control: protection model and policy language. In: Proceedings of the First ACM Conference on Data and Application Security and Privacy, CODASPY 2011, pp. 191–202. ACM, New York (2011). ISBN 978-1-4503-0466-5
13. Masoumzadeh, A., Joshi, J.: Osnac: an ontology-based access control model for social networking systems. In: Proceedings of the 2010 IEEE Second International Conference on Social Computing, SOCIALCOM 2010, pp. 751–759. IEEE Computer Society, Washington (2010). ISBN 978-0-7695-4211-9
14. Carminati, B., Ferrari, E., Heatherly, R., Kantarcioglu, M., Thuraisingham, B.: A semantic web based framework for social network access control. In: Proceedings of the 14th ACM Symposium on Access Control Models and Technologies, pp. 177–186. ACM, New York (2009). ISBN 978-1-60558-537-6

15. Cheng, Y., Park, J., Sandhu, R.: Attribute-aware relationship-based access control for online social networks. In: Data and Applications Security and Privacy XXVIII (2014)
16. Bruns, G., Fong, P.W.L., Siahaan, I., Huth, M.: Relationship-based access control: its expression and enforcement through hybrid logic. In: Proceedings of the Second ACM Conference on Data and Application Security and Privacy (CODASPY 2012), pp. 117–124. ACM, New York (2012)
17. Fong, P.W.L.: Relationship-based access control: protection model and policy language. In: Proceedings of the First ACM Conference on Data and Application Security and Privacy, CODASPY 2011, pp. 191–202. ACM, New York (2011). ISBN 978-1-4503-0466-5
18. Hu, H., Ahn, G.J., Jorgensen, J.: Multiparty access control for online social networks: model and mechanisms. IEEE Trans. Knowl. Data Eng. **25**(7), 1614–1627 (2013)
19. Stanford Large Network Dataset Collection. https://snap.stanford.edu/data/

Context-Based Data Analysis and Applications

A Semantic Approach in Recommender Systems

Huynh Thanh-Tai, Huu-Hoa Nguyen, and Nguyen Thai-Nghe(✉)

College of Information and Communication Technology, Can Tho University,
3/2 Street, Can Tho, Vietnam
httaik21@gmail.com, nhhoa@ctu.edu.vn, ntnghe@cit.ctu.edu.vn

Abstract. Recommender systems (RSs) suggest a list of items to users by using collaborative or content-based filtering. Collaborative filtering approaches build models from the user's past behaviors (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users, while content-based filtering approaches utilize attributes of the items to recommend additional items with similar properties. Although RS is aplied in many real systems, it has several problems that need to be solved, e.g., cold-start (new users or new items) problem, data sparse problem, and especially data scarcity problem since most of the users are not willing to provide their opinions on the items. In this work, we present a semantic approach to recommender systems, especially for alleviating the sparsity and scarcity problems where most of the current recommendation systems face. We create a semantic model to generate similarity data given an original data set, thus, the prediction model has more data to learn. Experimental results show that the proposed approach works well, especially for sparse data sets.

Keywords: Recommender systems · Ontology · Data scarcity · Semantic recommender systems

1 Introduction

Recommenders Systems (RS) help users to tackle the overloading problem by effectively presenting new contents adapted to the user's preferences. This system is a type of information filtering system. Indeed, RS is used to predict preference or rating that user may like or rate on an item which has not been seen in the past (item could be song, movie, video clip, paper, task, course… [15, 16]). For example, in an online shopping system such as Amazon, to maximize the user shopping capability, the system usually takes into account which user likes which item based on using the past behaviors of the user (these behaviors could be the user's rate, number of clicks, browsing time… on an item). Using these behaviors, the system can automatically predict the items, which the user may prefer and then recommend them to him/her [13].

However, current recommendation algorithms commonly suffer from data sparsity and scarcity problems where the models have not enough data to learn.

In this study, we present a semantic approach to recommender systems by using an Ontology model. From this semantic model, we can generate similarity data so that the prediction model has more data to learn. We step-by-step present how to build the

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 331–343, 2016.
DOI: 10.1007/978-3-319-48057-2_23

semantic model as well as how to integrate this model to the recommender systems. Finally, we evaluate the proposed approach by using several public data sets.

2 Related Works

Several works have been published in using semantic for recommender systems, however, each work has its own different purpose [14].

In [10], an ontological user profiling is employed for recommending academic research papers. While relationships are rich in semantics, the authors found that this approach has some limitations, as it fails to consider other types of concept relationships. The authors proposed a classification algorithm, based on the k-Nearest Neighbor classifier, that assigns topics to papers and the model predict topics for articles of similar neighbors.

The Hermes framework [6] offers a semantic-based approach for retrieving news which is directly or indirectly related to the concepts of interests from the domain ontology, which is called the knowledge base. The ontology consists of classes, e.g., Company and CEO, and the relationship between these classes, e.g., is CEOOf and has CEO. A concept is defined as either a class or an instance of a class, e.g., Company and Microsoft. The Hermes News Portal (HNP) is a Java implementation of the Hermes framework [6]. It allows the user to query the news and views the knowledge base. It uses Jena library for manipulating and reasoning with the OWL ontologies. For querying, it employs SPARQL and tSPARQL, which add time functionalities to the queries. The classification of the news is done using GATE [8] and the WordNet [3] semantic lexicon. Author used TF-IDF and Jaccard similarity for finding similar articles to recommend for users.

Ontologies can be used to improve content-based search, as seen in OntoSeek [7]. Users of OntoSeek navigate the ontology in order to formulate queries. Ontologies can also be used to automatically construct knowledge bases from web pages, such as in Web-KB [5]. Web-KB takes manually labelled examples of domain concepts and applies machine-learning techniques to classify new web pages. On the basis of automated capture of information as well as user interests serve the recommendation.

Another relevance system is CiteSeer [1], which uses content-based similarity matching to help searching for interesting research papers within a digital library. It used Jaccard coefficient and TF-IDF similarity.

Quickstep recommender system is proposed in [7, 11]. This system combines AKT ontology and OntoCoPI which has been shown that the system can reduce both the cold-start and interest-acquisition problems. Quickstep is a hybrid recommender system, addressing the real-world problem of recommending online research papers to researchers. User browsing behavior is unobtrusively monitored via a proxy server, logging each URL browsed during normal work activity. A nearest-neighbor algorithm classifies browsed URL’s based on a training set of labelled example papers, storing each new paper in a central database.

In [12], the research approaches a search architecture that combines classical search techniques with spread activation techniques applied to a semantic model of a given domain. Spread activation techniques are used to find related concepts in the ontology given an initial set of concepts and corresponding initial activation values. In this approach, the structure inherent in the basic ontology used clearly and automatically in training and updating the user profile. The authors use cosine similarity measure to compare the similarity of two documents. Research uses Protégé tools for building ontologies and uses SPARQL as a data query language.

Many researches have been focused on exploitation of semantics to improve the technical quality of their prediction. Most of them use the same approach semantics (semantic similarity) to enhance the performance of the approach based on the content, however, there are also some systems using collaborative filtering methods based on the user's profile stored in the Ontology. For example: ePaper [9] is a recommender system of scientific papers using the inheritance relationships of concepts in the domain to calculate the combination of concepts and describes an item concept which is collected from the user's preferences. FOAFing music project [4] is a recommender system using standard music vocabulary FOAF to set up user profiles and exploit the semantic description of songs, mainly the relationship of technology officers, to find similar songs listening habits of users to implement recommend.

Another recommender system uses semantic inference methods in both phases of the process that AVATAR [2] which is a recommender system for TV channels using back-propagation method (upward-propagation) and semantics similarity methods.

In this work, we focus on pre-learning step, which means that we propose building sematic model to generate more data before training the prediction model.

3 Proposed Approach

In this study, we propose a semantic approach to recommender systems. The purpose is to overcome the sparsity and scarcity problems in the current recommender systems. We create ontology models to store the items so that given an active item, the models can easily retrieve other semantically similar items which are already having user rating/feedback in the past.

The main idea of the proposed approach is given by a following example. Suppose in the past data we have "John like a car (e.g., Toyota Camry car) very much". In this example, the *user* is "John"; the *item* is "Toyota Camry car"; and the *rating* is "5 stars". Using the proposed semantic model, we can find the (top-N) highest similarity cars in the ontologies and assign them with the same user (John) and the same rating (5 stars). Thus, the original data set can be enriched after applying the proposed semantic model.

Moreover, this approach can also be used for the cold-star (new user) problem where the new user comes to the system in the first time and has no rating/feedback in the past, as introduced in the first case of the following figure.

3.1 Overall Model for Integrating Semantic into RS

An overall model for integrating semantic into recommender systems is proposed in Fig. 1. There are two possible contexts in this model.

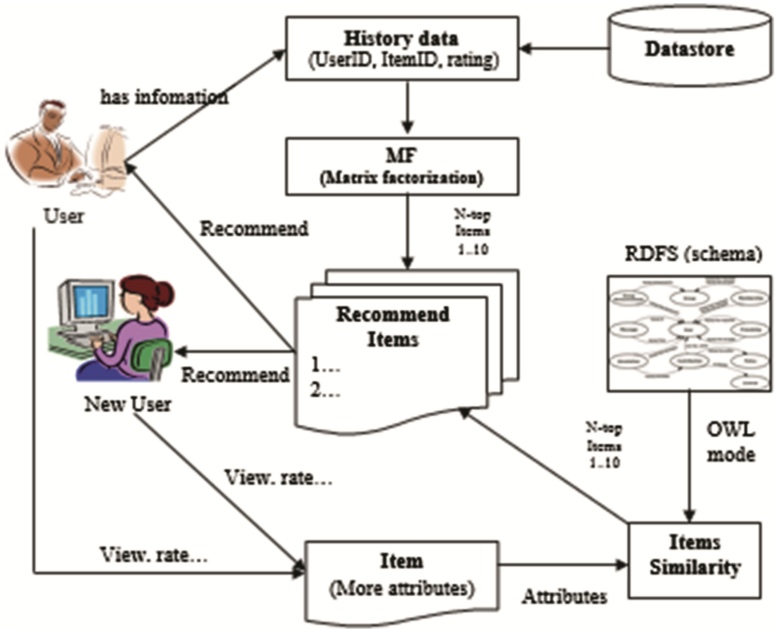


Fig. 1. Overall model

Context 1: For new users and guests. In this case, the users do not have any feedback/ rating information in the system, i.e. they have no “User-ID”, thus, the RS cannot recommend the items to those new users. However, in the proposed semantic model, we can provide recommendations for them easily by the following steps.

Step 1 (basic recommendation): The system recommends n-Top items for the user by using popular methods, e.g., most popular items, most buying items, most viewing items, or the new items, etc. This case depends on specific objectives of the application

Step 2: After the user chooses or views or even rates for an item, the system starts processing in the Ontology to find and recommend top-N items that are semantically similar to the active item (the item that is currently interacted with the user).

Context 2: For existing users. For the users who already have their information in the system, i.e., they had “UserID” and historical feedback (rating), the system recommends items to them as the following steps

Step 1: The system uses recommender algorithms, e.g., matrix factorization, to give n-Top items that the users might be interested.

Step 2: After the user chooses, views or rates for an item, the system start processing in the Ontology to find all items with semantically similar to the active item. Then, this result is combined with the results in the Step 1 to returns n-Top items for the current user. This can be considered as an ensemble approach.

For building the semantic model for RS, we will describe the structure of the Ontology as well as how to integrate it to the recommender system.

3.2 Ontology Structure

In order to store the items for processing as described in Fig. 1, the Ontology structure (RDFS Graph) is proposed Fig. 2. In this structure, the RDFS:Class is the class of all classes, RDFS:subClassOf transfers properties of the superclass rs:item into the new

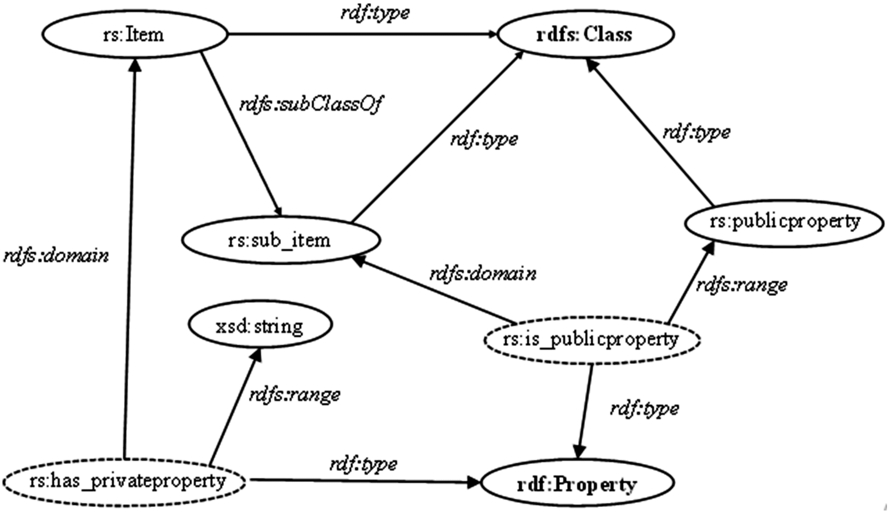


Fig. 2. Ontology structure

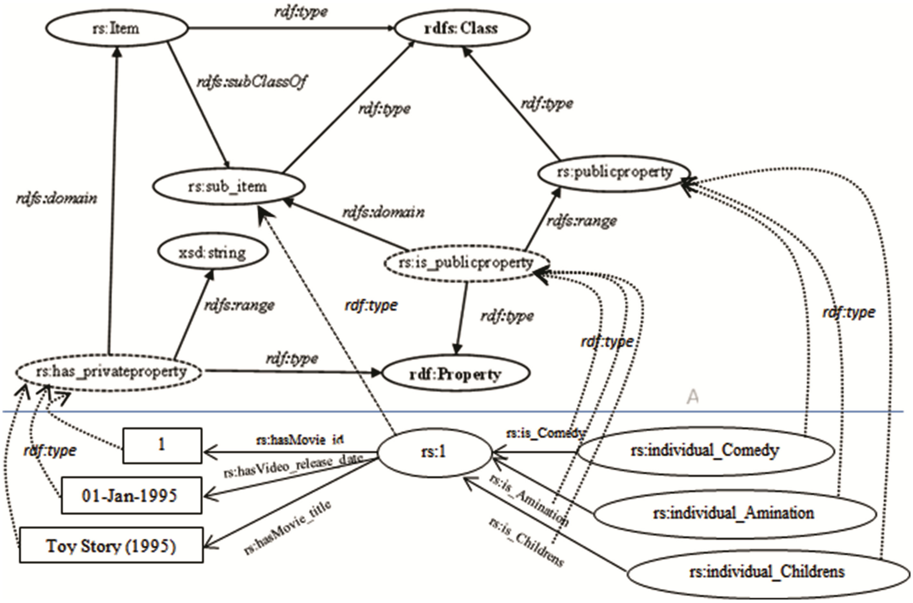


Fig. 3. An example of ontology structure

class rs:sub_item, RDFS:domain specifies domain of an attribute, RDFS:range specifies range of properties, rdf:property is the class of all properties, rdf:type specifies class of new classes or resources. In particular, for each instance of the Item class, we have many [privateproperty] and that is the same for each subclass of the Item class. This depends on the particular data set which is used to build the Ontology.

An example for this structure is presented in Fig. 3. This figure demonstrates for a record in MovieLens data set, which will be described in the experimental section, for example:

*(Movie_id=**1**, Movie_title=**Toy Story (1995)**, Video_release_date=**01-Jan-1995**, Movie_genre (attributes) = **{Comedy, Amination, Childrens}***

This record will then be mapped (dash arrows) to the ontology structure as seen in the below part of Fig. 2.

3.3 Building the Ontology

To be able to store and retrieve the items that the users rated in the past, we propose to build an Ontology as modeled in Fig. 2. This Ontology can be shared and reused of the knowledge of a domain. For building the Ontology, we perform the following steps:

Step 1. Create an empty OWLModel

Step 2. Using the structure in Fig. 2, we create a Class Item type, all classes [Privateproperties] and [Publicproperties]; note that [Privateproperties] belongs to Item domain, and [Publicproperties] is the [Class] type

Step 3. Create an instance of the class [Publicproperties], i.e., its corresponding Individuals

Step 4. Read each item from data. Each ItemID is an Individual (instance of the class Item) and the subclass is sub_item; Each item's properties is properties of Individual that has just created; Note that every [Publicproperties] is the property of Individual of class [Publicproperties] that was created in step 3;

3.4 Integrating Ontology into the Recommender Systems

To integrate Ontology into RS, the system needs to transfer the active ItemID which the current user is currently selecting/viewing to the RS-Integration as in Fig. 4. The main idea of this function is to combine the results from the recommender system with a list of items which have the highest semantic similarity retrieved from the Onlology for the active item.

In this study, we have used Jaccard coefficients similarity to calculate the similar between items which are stored in the ontology structure. Jaccard index, also known as Jaccard coefficients similarity, is a statistical coefficient was used to compare the similarities and diversity of the sample (sample sets). Jaccard coefficients similarity between samples A and B is defined as

$$\mathrm{J(A,B)} = \frac{|\mathrm{A} \cap \mathrm{B}|}{|\mathrm{A} \cup \mathrm{B}|} = \frac{|\mathrm{A} \cap \mathrm{B}|}{|\mathrm{A}| + |\mathrm{B}| - |\mathrm{A} \cap \mathrm{B}|}$$

```
procedure RS-INTEGRATION (userID, itemID, ktop)
// Let L[ktop] be return results of the traditional RS for userID
// Let M[ktop] and R[ktop] be return results of Similarity for itemID
R ← L
for k ← 1 to ktop do
    R[ktop + k] ← M[k]
end for
for j ← 1 to 2*ktop - 1 do
    biggest ← j
    for i ← j + 1 to 2*ktop do
        if R[i] == R[biggest] then remove(R, R[i]) //remove(from, what)
        if R[i]  > R[biggest] then
            biggest ← i
    end for
    R[j] ↔ R[biggest]
end for
return R[ktop]
end procedure
```

Fig. 4. Integrated procedure

3.5 Data Enrichment from Semantic Model

For tackle the problem of data sparsity and scarcity, we propose using semantic model to generate similarity data. The generation procedure is described in Fig. 5. This process is done through two main stages, as follows:

```
procedure SEMDATA($D^{rating}$, OWLmodel)
    // Let R be return results of Procedure
R ← null
size ← 0
for each item i from $D^{rating}$ do
    M returns results of Similarity for i in OWLModel
    n ← lengh of M
    for k ← 1 to n do
            R[size+k] ← M[k]
    end for
    size ← size + n
end for
    // remove duplicate items on R[size]
for i ← 1 to size - 1 do
    for j ← i + 1 to size do
        if  R[i]==R[j]  then  remove(R,R[j])  //remove(from, what)
    end for
end for
return R
end procedure
```

Fig. 5. Procedure for data enrichment

Stage 1: Preparation.

- Prepare data (collect the data from public data sets which have rating/feedback in the past). This data will be used for building semantic models.
- Sort the data (ASC/ DESC) by the UserID.
- Store the items to the Ontology structure as described in Sessions 3.2 and 3.3.

After data preparation phase is completed, we proceed to stage 2 for generating data.

Stage 2: Data enrichment. In this stage, we use the Ontology to retrieve all the rated items which are semantically similar to the given item.

After having enrichment data set, we can build the prediction model by using any method in recommender system. For testing purpose, in this work we have used the state-of-the-art method in Collaborative Filtering, which is Matrix Factorization [13, 14], however, other methods can also be applied.

Matrix Factorization (MF) is a technique that decomposes (approximately) a large matrix **X** into two smaller matrices **W** and **H**, such that **X** can be rebuilt from **W** and **H** as closely as possible [14], that means $\mathbf{X} \sim \mathbf{W}\mathbf{H}^{\mathbf{T}}$, as illustrated in Fig. 6. In this figure, $\mathbf{W} \in \Re^{|U| \times K}$ is a matrix where each row *u* is a vector with K latent factors which describe for user *u*; and $\mathbf{H} \in \Re^{|I| \times K}$ is a matrix which each column *i* is a vector with K latent factors that describe for an item *i* (please note that K<<|U| and K<<|I|)

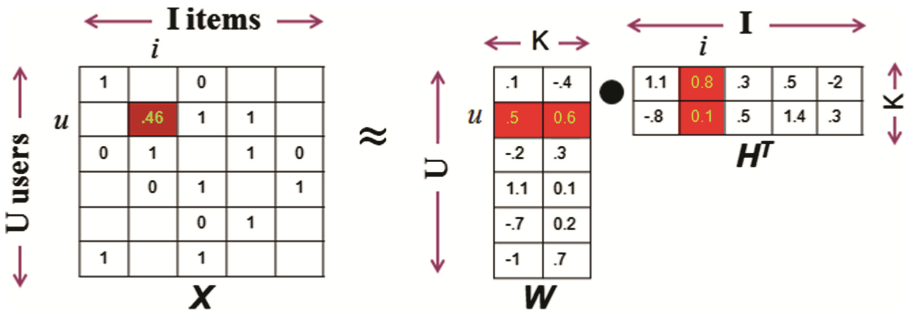


Fig. 6. Matrix factorization

Let w_{uk} and h_{ik} be the elements of **W** and **H**, then the rating by user *u* on item *i* is predicted by:

$$\hat{r}_{ui} = \sum_{k=1}^{K} w_{uk} h_{ik} = (WH^T)_{u,i} \tag{1}$$

The critical issue in the MF is to determine the values of two parameters **W** and **H**. These two parameters can be obtained by optimizing an objective function such as in the following (this optimizes for the squared error)

$$O^{MF} = \sum_{u,i \in D^{train}} (r_{ui} - \hat{r}_{ui})^2 = \sum_{u,i \in D^{train}} \left(r_{ui} - \sum_{k=1}^{K} w_{uk} h_{ik} \right)^2$$

After optimization process, we get the parameters **W** and **H**, then we can predict for the unseen data using formula (1). Please see more details in [13, 14].

4 Experimental Results

4.1 Data Sets

(a)Movielens data set (www.grouplens.org/datasets/movielens). This data set has many different versions: MovieLens 100 k, MovieLens 1 M and MovieLens 10 M. In this work, we have used the MovieLens 100 K which has 100,000 rating and it made by 943 users on 1682 films; each user rates for a films from 1 star (the worst) to 5 stars (the best). This data set is not sparse since we have counted that there are at least 100 ratings per user. The Movielens data set also has several attributes that can be used for building the Ontologies of the proposed semantic model, for example: movie title, IMDb URL, movie genre (action, adventure…), etc.

(b)MovieTweetings dataset. This dataset is available at github.com/sidooms/MovieTweetings. This is a dataset including the rating of 3,906 films collected on the Twitter website. MovieTweetings data is classified into the specific data segments: 10 k means previous data set collected 10,000 reviews; and 20 k means data collected before 20,000 reviews and similar data sets 200 k; This dataset also has several attributes, e.g., movie_title, movie_year, genre,… In this work, we have used MovieTweetings 10 K dataset; The original rating is assigned from 1 to 10.

(c)Restaurant & Consumer Data dataset (RCData). This dataset is stored at archive.ics.uci.edu/ml/machine-learning-databases/00232. It was collected from a part of the Restaurants recommender system rated by customers in Mexico City. Several attributes of this data set can be used to create Ontologies, such as Alcohol, Smoking_area, Accessibility, other_services, etc.

Table 1 presents the number of users in each data set on average. For the Movielens data set, it is not sparse while the Movietweetings and RCData are very sparse.

Table 1. Average number of ratings per user

Data sets	Number of users	Number of ratings	AVG rating
MovieLens	943	100,000	106
Movietweeting	3,794	10,000	3
RCData	138	1,161	8

4.2 Data Pre-processing

In order to store the Items in the Ontology structure in Fig. 2, the data needs to be pre-processed. For this purpose, we have presented Items as the following format

$$[\text{ItemID}\,]|[\,\text{privateproperties}\,]|[\,\text{publicproperties}]$$

For example: the MovieLens data is stored as the following:

movie_id|movie_title|video_release_date|IMDb_URL|unknown|Action| …

4.3 Evaluation Measures

In this work, we have evaluated the models using 3-folds cross validation. The popular measures in RS, which are Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), are used for assessment. These measures are defined as the following formulas:

$$\text{RMSE} = \sqrt{\frac{1}{|\text{D}^{\text{test}}|}\sum_{\text{u,i,r}\in \text{D}^{\text{test}}} \left(\text{r}_{\text{ui}} - \hat{\text{r}}_{\text{ui}}\right)^2}$$

$$MAE = \frac{1}{|\text{D}^{\text{test}}|}\sum_{\text{u,i,r}\in \text{D}^{\text{test}}} \left|\left(\text{r}_{\text{ui}} - \hat{\text{r}}_{\text{ui}}\right)\right|$$

Where $\text{D}^{\text{test}} \subseteq \text{U} \times \text{I} \times \text{R}$ is the test set; U: the set of users; I: the set of items; r_{ui}: actual value (rating); $\hat{\text{r}}_{\text{ui}}$: predicted value.

4.4 Experimental Results

The information from each data set before and after generating using semantic model is presented in Table 2. From these results, we can see that the MovieLens100 k data set after enrichment increases by 12.8 times compared to the original training data; For MovieTweetings and RCData data sets increase 41.34 and 4.15 times, respectively.

Table 2. Statistics on original data and enrichment data

Data sets / Number of Records	MovieLens 100k	MovieTweetings 10k	RCDdata
Original Data (OldData)	100,000	10,000	1,161
Test Data	29,942	2,569	336
Train Data	70,058	7,431	825
Enrichment Data (SemData)	897,098	307,158	3,421
Increment ratio	**12.8**	**41.34**	**4.15**

Experimental results using RMSE and MAE on 3 data sets are presented in Figs. 7 and 8. In these results, we have used the same well-known technique in RS, which is Matrix Factorization [13, 14], on the original data (denoted as OldData) and enrichment data by using semantic model (denoted as SemData).

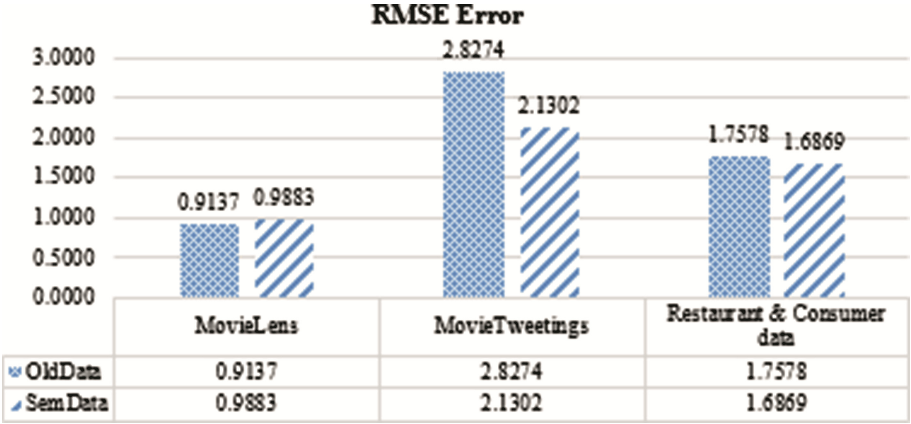


Fig. 7. RMSE results

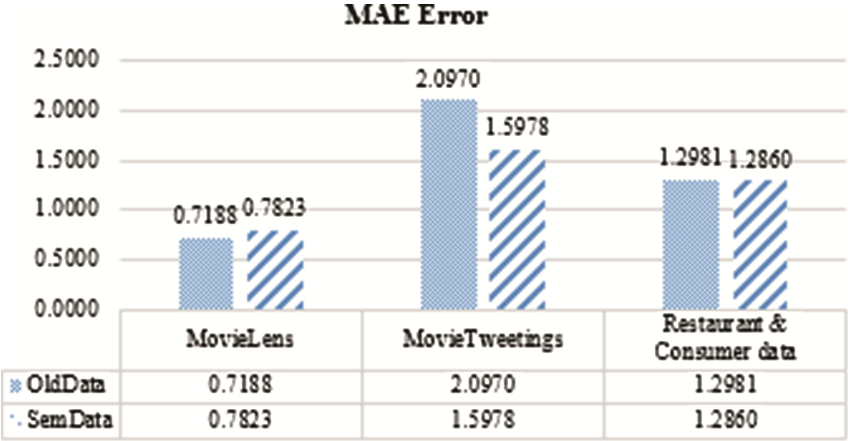


Fig. 8. MAE results

From these results, we can see that by using semantic model to generate more data, the prediction model has more data to learn, thus, the model can reduce the prediction errors on both RC-Data and Movie Tweetings data sets, especially the error is significantly reduced on Movie Tweetings data set.

However, using enrichment data on Movielens data set, it does not help but producing negative results. The reason for this case is that the Movielens data set is not sparse (more than 100 ratings per each user, on average) thus, after generating more data, the model gets over-fitting. These results help us validating that the proposed semantic recommendation approach can be used for tackling the sparsity and scarcity problems.

5 Conclusions

In this work, we have proposed a semantic approach to recommender systems, especially for alleviating the sparsity and scarcity problems where most of the current recommendation systems face. We create a semantic model to generate similarity data given an original data set, thus, the prediction model has more data to learn. Experimental results show that the proposed approach works well, especially for sparse data sets.

Using semantic model for tackling the cold-start (new user/ new item) problem could be a potential topic for future work.

References

1. Bollacker, K.D., Lawrence, S., Giles, C.L.: An autonomous web agent for automatic retrieval and identification of interesting publications. In: Proceedings of the Second International Conference on Autonomous Agents, Minneapolis MN, USA (1998)
2. Blanco-Fernández, Y., et al.: A flexible semantic inference methodology to reason about user preferences in knowledge-based recommender systems. Knowl. Based Syst. **21**(4), 305–320 (2008)
3. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
4. Celma, O., Serra, X.: FOAFing the music: bridging the semantic gap in music recommendation. Web Seman. Sci. Serv. Agents World Wide Web **6**(4), 250–256 (2008)
5. Craven, M.D., Freitag, D., McCallum, D., Mitchell, A., Nigam, K., Slattery, S.: Learning to extract symbolic knowledge from the world wide web. In: Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-1998) (1998)
6. Frasincar, F., Borsje, J., Levering, L.: A semantic web-based approach for building personalized news services. Int. J. E-Bus. Res. **5**(3), 35–53 (2009)
7. Guarino, N., Masolo, C., Vetere, G.: OntoSeek: content-based access to the web. IEEE Intell. Syst. **14**(3), 70–80 (1999)
8. Cunningham, H.: GATE: a general architecture for text engineering. Comput. Humanit. **36**, 223–254 (2002)
9. Maidel, V., Shoval, P., Shapira, B., Taieb-Maimon, M.: Evaluation of an ontology-content based filtering method for a personalized newspaper. In: RecSys 2008 Proceedings of the 2008, pp. 91–98 (2008)
10. Middleton, N., Shadbolt, R., Roure, D.C.D.: Ontological user profiling in recommender systems. ACM Trans. Inf. Syst. **22**(1), 54–88 (2004)
11. Anh-Thu, L.N., Nguyen, H.-H., Thai-Nghe, N.: A Context-aware implicit feedback approach for online shopping recommender systems. In: Nguyen, N.T., Trawinski, B., Fujita, H., Hong, T.-P. (eds.) ACIIDS 2016. LNCS, vol. 9622, pp. 584–593. Springer, Heidelberg (2016). doi: 10.1007/978-3-662-49390-8_57
12. Vadivu, G., Hopper, W.: Ontology mapping of indian medicinal plants with standardized medical terms. J. Comput. Sci. **8**(9), 1576–1584 (2012)
13. Thai-Nghe, N.: An introduction to factorization technique for building recommendation systems. J. Sci. Univ. Da Lat **6/2013**, 44–53 (2013). ISSN 0866-787X

14. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. IEEE CS **42**(8), 30–37 (2009)
15. Thai-Nghe, N., Horváth, T., Schmidt-Thieme, L.: Personalized forecasting student performance. In: Proceedings of the 11th IEEE International Conference on Advanced Learning Technologies (ICALT 2011), pp. 412–414. ISBN: 978-1-61284-209-7. IEEE Xplore (2011)
16. Thai-Nghe, N., Drumond, L., Horváth, T., Schmidt-Thieme, L.: Using factorization machines for student modeling. In: Proceedings of FactMod 2012 at the 20th Conference on User Modeling, Adaptation, and Personalization (UMAP 2012), vol. 872, CEUR-WS, ISSN: 1613-0073 (2012)

Automatic Extraction of Semantic Relations from Text Documents

Chien D.C. Ta(✉) and Tuoi Phan Thi

Faculty of Computer Science and Engineering,
Ho Chia Minh City University of Technology, Ho Chia Minh City, Vietnam
{chientdc,tuoi}@cse.hcmut.edu.vn

Abstract. Ontologies play an important role of the applications in recent years, especially on the Semantic Web, Information Retrieval, Information Extraction, and Question Answering. Ontologies are used in order to specify the knowledge that is exchanged and shared between the different systems. Ontologies define the formal semantics of the terms used for describing data, and the relations between these terms. Hence, one of the important steps for building the domain specific ontology is to construct the semantic relations among the terms of the ontology. This paper represents our proposed method for automatic semantic relation extraction from text documents of the ACM Digital Library. A random sample among 170 categories of ACM categories is used to evaluate our proposed method. Results generated show that our proposed method achieves high precision.

Keywords: Semantic relations · Computing domain ontology · Information extraction

1 Introduction

Ontology is defined as formal and explicit specification of shared conceptualization [1]. It represents knowledge in structured form suitable for inference and reasoning over knowledge. However, the development of domain ontology is not a trivial task and consumes important resources in term of time and mone. In order to build ontologies, especially the domain specific ontologies, we must explore many resources related to domain specific for the extraction of concept instances and semantic relations. Malaisé [2] used lexical syntactic patterns to detect semantic relations between the main terms of definition in order to help terminologist build structured terminology following these relations. Their research achieved good results. With the dramatic increase of data, the automatic semantic relation construction from text documents plays an important role in semantic applications. Numerous studies and tools can already be found in the scientific literature. They include Gate, Termine, Stanford CoreNLP, etc. Generally, they can detect and extract semantic relations on the different domains. Unfortunately, existing tools suffer from two main limitations. The first is that the precision factor is not high since they can use in any domain, not only focus on the Computing domain. The second limitation is that they cannot detect semantic relations, such as synonyms, hyponyms, and hypernyms among the instances. Our goal is to automatically identify the semantic

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 344–351, 2016.
DOI: 10.1007/978-3-319-48057-2_24

relations that might be found in text documents of the ACM Digital Library. Afterward, we extract these relations in order to enrich domain specific ontology. This ontology can be used in many applications, such as Information Retrieval, Information Extraction, Question answering focusing on the Computing domain. For this purpose, we propose a methodology, which combine Natural Language Processing (NLP) and statistical method. In order to evaluate this methodology, we use three measures, namely Precision, Recall and F-Measure. The evaluation results prove the effectiveness of the proposed methodology.

Our key contribution are as follows: (i) we propose a hybrid approach combining NLP and statistical method for semantic relation extraction from text documents focusing on the Computing domain; (ii) the semantic relations is not only synonym, hyponym, hypernym relations, but also the other relations such as IS-A, PART-OF, MADE-OF, ATTRIBUTE-OF, etc.

The rest of this paper is organized as follows: Sect. 2 examines related work; Sect. 3 introduces the proposed methodology; Sect. 4 illustrates the experimental results; Sect. 5 discusses conclusions and future work.

2 Related Work

Information extraction is an important research topic in NLP, especially relevant to extracting semantic-oriented data. Gomez et al. [3] built a semantic interpreter to assign meaning to the grammatical relations of the sentences when they constructed a knowledge base about a given topic. Kongkachandra et al. [4] proposed semantic based keyphrase recovery for domain-independent key-phrase extraction. In this method, he added a key-phrase recovery function as a post process of the conventional key-phrase extractors in order to reconsider the failed key phrases by semantic matching based on sentence meaning. Zoudong et al. [5] proposed novel tree kernel-based method with rich syntactic and semantic information for the extraction of semantic relations between named entities. Abacha et al. [6] built a platform MeTAE (Medical Texts Annotation and Exploration). This system allows the extracting and annotating of Medical entities and relationships from Medical text. He relied on linguistic patterns to detect the semantic relations in medical text files. Jayatilaka et al. [7] constructed ontology from Web pages. He introduced web usage patterns as a novel source of semantics in ontology learning. The proposed methodology combines web content mining with web usage mining in the knowledge extraction process. Li et al. [8] extract semantic relations between Chinese named entities based on semantic features and the Vector Space Model (VSM).

Those researchers attempt were meant to identify the semantic relations from web documents or text documents in order to construct the ontology. They either used NLP processing techniques, the statistical method, or the machine learning approach in the ontology learning process. However, the precision and the recall of those approaches are not high and the type of semantic relations is limited. Our research combines NLP with statistical for identifying and extracting the semantic relations among instances of domain specific ontology.

3 Automatic Semantic Relation Extraction from Text Documents

3.1 Computing Domain Ontology (CDO)

Formally, an ontology can be defined as the tuple [9]:

$$O = (C, I, S, N, H, Y, B, R)$$

Where,

C, is set to consist of classes. In this ontology, C represents categories of computing domain (e.g., "Artificial Intelligent, hardware devices, NLP" ∈ C); I is set of instances belong to categories. In this ontology, set I consists of computing vocabulary (e.g., "robotic, Random Access Memory" ∈ I); $S = N^S \cup H^H \cup Y^H$ is the set of synonyms, hyponyms and hypernyms of instances of set I; N = NS is set of synonyms of instances of set I; H = HH is set of hyponyms of instances of set I; Y = YH is set of hypernyms of instances of set I; B = {belong_to (i, c) | i ∈ I, c ∈ C} is set of taxonomy orders between concepts of set C and instances of set I and are denoted by {belong_to (i, c) | i ∈ I, c ∈ C} meaning that i belongs to category c; R = {rel (s, i) | s ∈ S, i ∈ I} is the set of relationships between terms of set S and instances of set I and are denoted by {rel (s, i) | s ∈ S, i ∈ I} meaning that s is relationship with i.

The purpose of this paper is to automatically extract the relationships in the set R. This process includes two steps, the first step is to identify the instances and the second step is to identify the types of relationships between them. In addition, all concepts and instances of this ontology focus on the Computing domain; therefore, this ontology is called as Computing Domain Ontology (CDO). The overall hierarchy of CDO is shown in Fig. 1.

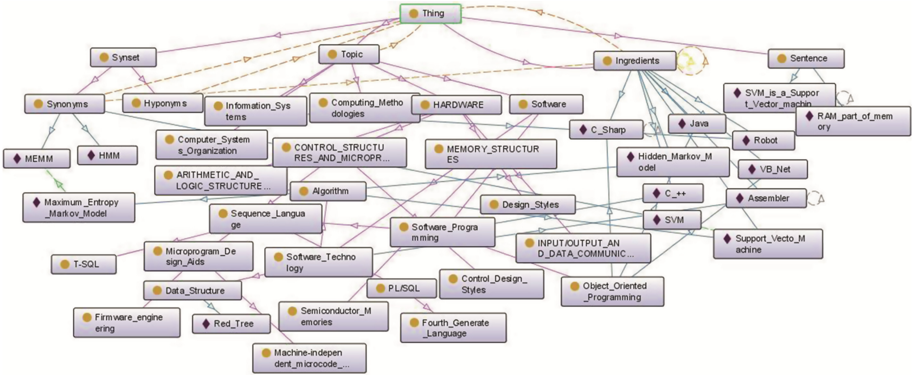


Fig. 1. CDO hierarchy is presented by Protégé

3.2 Identifying the Semantic Relations Among the Instances of CDO

Definition 1. Semantic relations – Semantic relation is the primary building block of many NLP applications, such as Ontology Learning, Question Answering system. Different types of semantic relations are mentioned in literature, such as hypernym, hyponym, synonym [10].

After selecting the instances from the above step, the next step is to identify the types of semantic relations among the instances. Besides the semantic relations such as synonyms, hyponyms and hypernyms relations, in this paper, we introduce some others, namely IS-A, PART-OF, MADE-OF, DELIMITED-BY, TAKES-PLACE-IN, ATTRIBUTE-OF, RESULT-OF, AFFECTS.

IS-A: this generic-specific relation reflects hierarchical inheritance in network of concepts. All entities are categorized as instances of a particular class. Class can become instances of a particular class. Thus, any concepts can be linked to its immediate superordinate concept. For example, Random Access Memory (concept) is a core memory (concept) in the computer.

PART-OF: this relation also reflects the hierarchical structure of the domain. This relation directly refers to the parts of each concept in a sentence. For example, Random Access Memory (ROM) (concept) is part of memory (concept).

MADE-OF: this relation links to concepts, which made of material concepts. For example, Integrated Circuit (IC) Chip can be made of a semiconductor material.

DELIMITED-BY: this relation marks the boundaries, dividing one concept from another. This is a domain-specific relation, mainly for the concepts, which are belonged to different topic in the field of Information technology. This relation usually is represented by a number of verbs, such as *include, delimit, limit, circumscribe, restrict,* etc… For example, the processing of computer is restricted by CPU, RAM.

TAKES-PLACE-IN: this relation describes the context of processes, which are related to spatial and temporal dimensions. A number of verbs represent this relation, such as *happen, occur, take place in*, etc. For example, in order to tackle the conflict of process, time scheduling takes place in the Operating System.

ATTRIBUTE-OF: this relation is only useful for concepts designated by specialized adjectives, such as *strong*, powerful, etc., or nouns that define the properties of other concepts. For example, these router devices are powerful and useful in network.

RESULT-OF: this relation is relevant to either processes or entities that are derived from other processes. For example, as a result of the inconsistency, this file is considered corrupted.

AFFECTS: this relation, along with RESULT-OF, are crucial semantic relations in the knowledge base for both can relate all kinds of concepts in the ontology.

WordNet is used in order to construct the synonym, hyponym and hypernym relations. The others are constructed based on SLDP. We propose two algorithms for identifying the semantic relations as follows.

Algorithm 1. Semantic relation extraction based on WordNet

```
Input: instances[]
Output: List of Synonyms, hyponyms, hypernyms of intances
For each instance in intances[]

|      Begin
|         Synonyms   = QueryIntoWordNet(instance)
|         Hyponyms   = QueryIntoWordNet(instance)
|         Hypernyms = QueryIntoWordNet(instance)
|      End
End for
```

Example 1. The synonyms, hyponyms and hypernyms of the instances are constructed after applying algorithm 1 as shown in Table 1.

Table 1. The synonym, hyponym and hypernym relations of the instances

Instances of Ingredient layer	Synonyms	Hyponyms	Hypernyms
NLP	Natural Language Processing		Informatics', information processing
Data structure		Hierarchical structure	Organization, system
Computer Network		Internet, intranet, WAN	Electronic network
RAM	Random Access Memory	Core memory	Volatile storage

Algorithm 2. Constructing the other semantic relations

```
Input: Sentence, template_verb[]
Output: Semantic relations such IS-A, PART-OF, MADE-OF
/* Sentence which contents the instances is analyzed by SLDP */
Dependency_relation[] =
                   Analysis_Sentence_bySLDP(Sentence)
For each relation in Dependency_relation[]
|   If (verb ∈ relation and verb ∈ template_verb[])
|      identify_type_semantic_relation(verb)
|   end If
|   /* Refining sentence based on SLDP */
|   Semantic_relation = Sentence_refine(Sentence)
|   Insert_CDO(Semantic_relation)
End For
```

In the algorithm 2, the function *sentence_refine(sentence)* will refine the sentence by SLDP. It means that we eliminate the unnecessary words in the sentence based on the dependency tree generated by SLDP. Some examples are shown in Table 2.

Table 2. Examples of eliminating unnecessary words.

The Original sentences	Sentences refined using SLDP
COBOL is not a popular programming language in recent years.	COBOL is not a popular programming language.
Oracle database is one of the Relational Database Management System.	Oracle database is Relational Database Management System.
In my opinion Java Language is an object oriented programming language	Java Language is an object oriented programming language

The semantic relations are shown in Fig. 2.

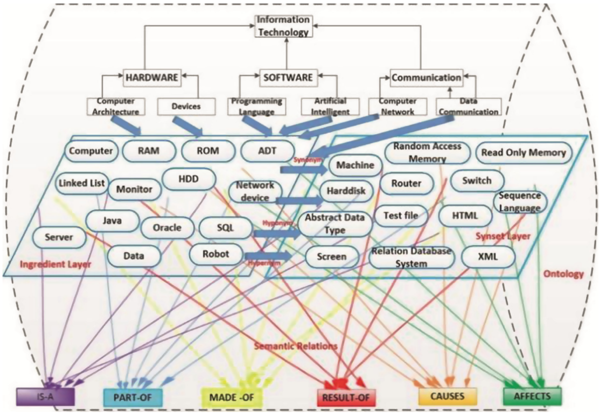


Fig. 2. The semantic relations are represented in CDO.

4 Experimental Results

4.1 Comprehensive Evaluation Method

The proposed approach is evaluated by three measures, namely Precision, Recall and F-measure. These measures are calculated by each category in CDO as below:

$$P(C_i) = \frac{Correct(C_i)}{Correct\left(C_i\right) + Wrong(C_i)} \tag{1}$$

$$R(C_i) = \frac{Correct(C_i)}{Correct(C_i) + Missing(C_i)} \tag{2}$$

$$\mathrm{F-Measure}(C_i) = 2\frac{\mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{3}$$

Where C_i represents a category in CDO and correct, wrong, missing represent the number of correct, wrong, missing, respectively. We pick a random four categories from ACM Digital Library as below:

- One corpus with 100 papers in Assembly Language category; One corpus with 100 papers in Database System category; One corpus with 100 papers in Software Engineering category; One corpus with 100 papers in NLP category.

The experimental results are shown in Table 3.

Table 3. Comprehensive evaluation method

Categories	Precision (%)	Recall (%)	F-measure (%)
Assembly language	85.61	56.14	67.81
Software engineering	93.47	81.16	86.88
Database system	82.53	71.07	76.37
NLP	89.18	85.31	87.20

4.2 Comparative Evaluation Method

We use Stanford CoreNLP [11] for comparative evaluation method. Stanford CoreNLP is a tool for extraction of instances and relations among instances from text documents. Stanford CoreNLP supports the API functions to develop the applications related to NLP. In order to compare the results, we pick a random two categories from the ACM Digital Library as below:

- One corpus with 100 papers in Assembly Language category.
- One corpus with 100 papers in Database System category.

The experimental results are shown in Table 4 as below.

Table 4. Comparative evaluation method

Categories	Precision (%)	Recall (%)	F-measure (%)
Assembly language	85.61	56.14	67.81
Database system	82.53	71.07	76.37
Assembly language (CoreNLP)	69.46	62.13	65.59
Database system (CoreNLP)	62.37	73.75	67.58

The scores reported in Table 4 reveals that the precision and F-measure of our proposed approach are higher than the CoreNLP tool but the recall is lower. Generally, our proposed method outperforms the Stanford CoreNLP tool. In the future word, we will enrich CDO from the other corpora to improve the recall factor.

5 Conclusion

Our experiment extracted the semantic relations from text documents based on WordNet and the NLP tools such as OpenNLP, SLDP in order to build an ontology on the Computing domain. After using the OpenNLP tool to chop sentences, we apply SLDP for POS tag and using Information Gain to filter instances before inserting to Computing Doman Ontology. We then identify the type of semantic relations among the instances of the sentences in text documents. In order to have the semantic relations, we refined the sentences using the SLDP tool. Therefore, our experimental results have high precision and high recall. Overall scores are computed based on three measures, namely Precision, Recall and F-measure.

References

1. Gruber, T.: A translation approach to portable ontology specifications. Knowl. Acquisition **5**(2), 199–220 (1993)
2. Malaisé, V., Zweigenbaum, P., Bachimon, B.: Detecting semantic relations between terms in definitions. In: Proceedings of the 3rd International Workshop on Computational Terminology (CompuTerm 2004) (2004)
3. Gomez, F., Segami, C.: Semantic interpretation and knowledge extraction. Knowl.-Based Syst. **20**(1), 51–60 (2006)
4. Kongkachandra, G., Chamnongthai, K.: Abductive reasoning for keyword recovering in semantic-based keyword extraction. In: Proceedings of the 5th International Conference on Information Technology: New Generations. IEEE (2008)
5. Zhou, G., Qian, L., Fan, J.: Tree kernel-based semantic relation extraction with rich syntactic and semantic information. Inf. Sci. **180**(8), 1313–1325 (2009)
6. Abacha, A.B., Zweigenbaum, P.: Automatic extraction of semantic relations between medical entities- a rule based approach. J. Biomed. Semant. **2**(5), 1 (2011)
7. Jayatilaka, A.D.S.: Knowledge extraction for semantic web using web mining. In: The International Conference on Advances in ICT for Emerging Regions (ICTer 2011). IEEE (2011)
8. Li, H., Wu, X., Li, Z., Wu, G.: A relation extraction method of chinese named entities based on location and semantic features. Appl. Intell. **18**(1), 1–14 (2012)
9. Zhang, L.: Ontology based partial building information model extraction. J. Comput. Civ. Eng. **27**, 1–44 (2012)
10. Ittoo, A., Bouma, G.: Minimally-supervised extraction of domain-specific part–whole relations using Wikipedia as knowledge-base. Data Knowl. Eng. **85**, 57–79 (2013)
11. Stanford CoreNLP–a suite of core NLP tools. Stanford University. http://stanfordnlp.github.io/CoreNLP/. Accessed 2016
12. OpenNLP. https://opennlp.apache.org/
13. Stanford Lexical Dependency Parser. http://nlp.stanford.edu/software/lex-parser.shtml

Emerging Data Management Systems and Applications

The Present and Future of Large-Scale Systems Modeling and Engineering

Dirk Draheim(✉)

Large-Scale Systems Group,
Tallin University of Technology, Tallinn, Estonia
dirk.draheim@ttu.ee

Abstract. Today's society and organizations rely on large-scale and ultra-large scale IT systems. Large-scale IT systems drive social and organizational change. We find them as the backbone of what we call the digital society, the digital economy, the fourth industrial revolution and so forth. Large scale-systems show as systems of systems or IT system landscapes. They show as data-intensive systems, workflow-intensive systems, massively resource-intensive systems, highly distributed systems. How to deal with the complexity of large-scale systems? How to approach architecture, design, realization and management of large-scale systems in systematic and rigorous ways? In this talk we attempt a foundational review of modeling and engineering techniques available for large-scale systems. From this, we try to understand possible pathways, both short-term and long-term, of large-scale systems modeling and engineering.

Keywords: Ultra large-scale systems · Big data · Cloud computing · e-government · e-governance · IT system landscapes

1 Introduction

Today's society and organizations rely on large-scale and ultra-large scale IT systems. Large-scale IT systems drive social and organizational change. We find them as the backbone of what we call the digital society, the digital economy, the fourth industrial revolution and so forth. How to deal with the complexity of large-scale systems? How to approach architecture, design, realization and management of large-scale systems in systematic and rigorous ways? In this talk we aim at a foundational review of modeling and engineering techniques available for large-scale systems. We do so, by discussing three different, fundamental perspectives on the way we perceive IT-enabled systems, see Fig. 1, i.e., a system-theoretic perspective, a normative perspective and the engineering perspective, which is, actually, our genuine perspective. Altogether, this is a preliminary attempt to understand IT-based systems through high-level perspectives. From this, we try to understand possible pathways, both short-term and long-term, of large-scale systems modeling and engineering.

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 355–370, 2016.
DOI: 10.1007/978-3-319-48057-2_25

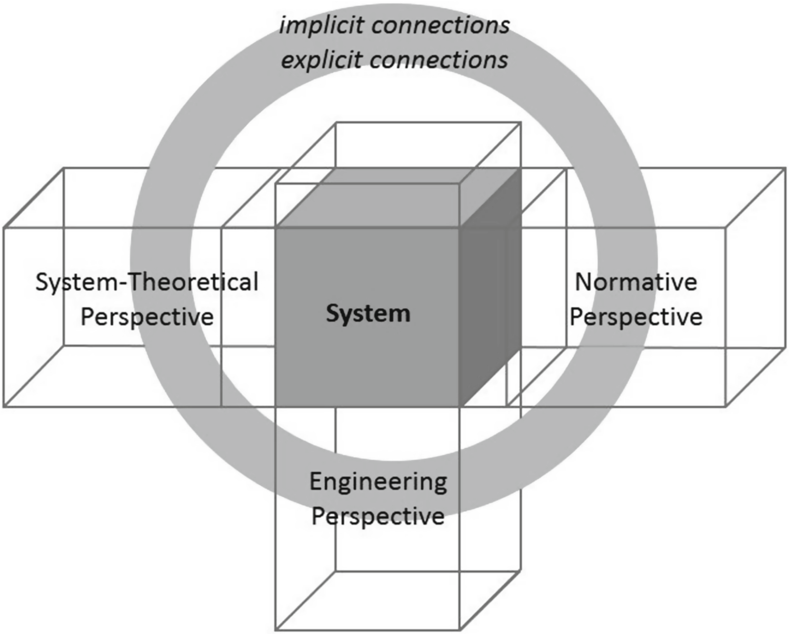


Fig. 1. A preliminary attempt to understand IT-based systems through high-level perspectives.

A first attempt to characterize large-scale systems is merely in terms of their way-above average large size in any relevant dimension like line of code, number of stakeholders, amount of managed data, degree of distribution and so forth. Actually, a driving question that led to the important investigation of ultra-large scale systems in [57] has been how to deal with IT systems that consist of a billion lines of code. However, see again [57], merely size is not always what makes a large-scale system. We are interested in systems of system and IT system landscapes. For a mature characterization of what systems of systems are about see [45]. Systems of system consist of components that are developed and managed separately. Typically, they undergo a steady extension and improvement of their functionality. Furthermore, typically their functionality is somehow more than the sum of their parts. It is not our concern in this talk to define what a large-scale system actually is. However, we are not interested in any kind of large IT system or IT-based system, but those that we find inside large organizations to manage people, processes and resources, systems in cross-organizational scenarios, e-Commerce, e-Health [55] and, last but not least, in e-Government [53] and e-Governance [54]. Certain points are important for us. Large-scale systems are highly complex. They are socio-technical systems. The environments in which they are embedded matter and are themselves highly complex. Large-scale systems challenge us. Standard techniques of modeling and engineering are not sufficient to address them properly. Technologically, large-scale systems show as data-intensive systems [13,22], workflow-intensive systems, massively resource-intensive systems and highly distributed systems.

The engineering approach to systems is a very successful one. The engineering principle is about a triad of rigorous analysis, rigorous design and implementation on the basis of best available technology. It was the credo of the early software development community [15,49] to turn the art of programming into an engineering discipline and since then we have seen tremendous achievements in software engineering methodology. But in understanding large-scale systems, when it comes to socio-technical systems, there are other important perspectives, see Fig. 1. We look at the system-theoretic perspective and at a normative perspective. The choice of these perspectives and their characterization in this talk are preliminary and in a sense arbitrary. It is the attempt to come closer to the complexity and nature of large-scale systems, in service of improving model and engineering techniques for them. The perspective yields a broad model. Other perspectives are possible and, in particular, more refined ones. However, the chosen perspectives are not completely arbitrary, actually, they stem from the midst of how we encounter, of how deal with large-scale systems. The perspectives are no silos. They overlap and they are implicit and explicit connections between them that are worth to investigate.

We proceed as follows. In Sect. 2 we discuss the system-theoretic, normative and engineering perspective on large-scale systems. In Sect. 3 we try to derive possible pathways for modeling and engineering techniques and technologies for large-scale systems. The discussion is necessarily biased as it reports, at least in parts, on current and future work. We discuss related work throughout the paper and finish with a conclusion in Sect. 4.

2 IT System Engineering Perspectives

2.1 On the System-Theoretical Perspective

The nature of system-theoretic thinking is rather deductive. There is a general idea from which systems are understood. This idea is a spark, it can stem from inspiration, also, from a deep insight. A system theoretic model must not be mixed with what we call a meta model in information system science. It is not a blueprint for modeling systems. Rather, it models systems-related phenomena at a high level of abstraction. Concrete system theories vary in the phenomena they address and in how domain-specific they are.

With cybernetics Norbert Wiener strived for a theory of messages in their role of controlling systems, i.e., any kind of system ranging from machinery to society. Cybernetics builds onto insights from several fields, i.e., electrical engineering, psychology, biology, computers. In [62] Norbert Wiener characterizes cybernetics as the science of “control and communication in the animal and the machine”. In [61] Norbert Wiener exploits cybernetics thinking as a device to reflect upon society and societal change. The viable system model of Stafford Beer is a cybernetic model. The viable system model understands a system as acting in and reacting to a changing environment. It explicitly draws an analogy to biology and organisms [3]. A viable system is a goal-oriented, recursive feedback-control system. Its goal is to survive in its environment, and as a viable system it is successful with respect to this goal. The viable system model details out the several essential components of a viable system, explains their roles, functionalities and their interplay. The model claims some universal truth. In that sense, the analogy to biological organisms is more than just a metaphor, more than just a didactical device. Rather, the analogy is part of the justification of the model. Over and over again, it is very instructive, how components of today's organization can be mapped to components of the viable system model. These mappings are meaningful and can be exploited to derive further insights or claims concerning the investigated organization.

The viable system model itself is universal. Not only it strives for explaining the single organization and its part. Rather, it has been applied to whole societies. In the 1970s Salvador Allende asked Stafford Beer for consulting in the elaboration and establishment of a real-time command economy in Chile. The project had the name Cybersyn and the system has actually set into action [10]. A kind of internet has been established in these early days in Chile. The so-called Cybernet was based on telex-machines and connected to a central opsroom in Santiago de Chile. There was a statistical modeling system Cyberstride [10] and an economic simulation system CHECO (Chilean economic simulator) [47]. The whole system was designed after the viable system model. So, the viable system model is not mere fiction. And again, it comes to our minds, when we look at systems like the Blackrock Aladdin system. The Aladdin system is a collective intelligence system that connects experts with big data and supercomputing capabilities. In 2013 the system held information about $15 trillion of financial assets, at this point in time this was 7 % of the world's $225 trillion of financial assets [29]. The system connects thousands of world-wide experts from the financial sector and the several industrial domains in real-time to enable risk assessment. As such, it is a huge monitoring and decision-support system. Of course, the Aladdin system operates in the free market and the control loop is not completely closed as in the Cybersyn vision.

System theoretic thinking can show in models that are much more domain-specific, or let us say more narrow to the kind of systems they investigate. Let us have a look at the principal-agent model of Jensen and Meckling [42]. It is explicitly a theory of the firm. As such, it is a high-level theory. It is about the ownership structure of a company and builds upon elements of agency theory, theory of property rights and theory of finance. A system consists of actors, some of them are principles, some of them are agents. Principals hire agents to perform tasks. Both principals and agents have self-interest and try to maximize their benefits. A strength of the model is in the analysis of costs and their interrelationships. For example, agency costs are explained to consist of incentives, monitoring costs, and bonding costs. We already see, how the model helps us in understanding organizational control switches. Actually, the model can help us in analyzing and predicting people's behavior in a firm.

2.2 On the Normative Perspective

The world of normative thinking is the world of standards and regulations. It is the realm of processes, responsibilities, accountabilities, key performance indicators, policies, laws and contracts. As opposed to system theory, people in practice necessarily come in touch with the normative perspective. Regulations must be respected and standards help. Often, standards are authoritative, then, they are de-facto the only way to fulfill regulations. Standards are dually experienced as theoretical and practical at the same time. If you conduct a typical ISO9000 [41] project in your company, you will often receive comments that the standard is not practically useful, merely theoretical stuff by bureaucrats. On the other hand, the project might be practically just necessary. Maybe you need it to be fit for your customers. Maybe these potential customers even formally require the fulfillment of this standard. On the other hand, you might need to conduct such a project to drive change in your organization. Normative thinking also establishes system theories, at least a system model. As such, it is more hands on, i.e., the model is built only in so far as it is in service of the normative needs of an organization. Therefore, the models that we find in the normative environment are less founded and elaborated. However, it is often possible to map them to a system theory.

Actually, the normative environment is wider than the concrete tools and driving forces, i.e., standards and regulations discussed so far. It is about the established distribution of decision making, order and control, i.e., about the distribution of power in a system. With respect to single organizations this perspective is therefore about organizational culture [59]. For single organizations, it often turns out that it is most appropriate to understand them as recursive feedback control systems [17] which leads us back to the viable system model. Larger systems, i.e., markets, economies and society are far more complex and also need other system theories.

2.3 On the Systems Engineering Perspective

The engineering approach to systems is about rigorous analysis, design and construction. As such, it is forward-thinking and experiences the world as a sequence of practical challenges that need to be resolved. But engineering thinking has also its limitations. The systems engineering perspective is very well aware of the normative perspective. Its issues are often perceived as requirements, side conditions and challenges. Sometimes, the true nature of the normative environment, i.e., establishment and governance of an organization, is not fully understood or even neglected by the engineering perspective. Similarly, the engineering approach cannot exploit deeper insight and ideas about a system than those it can achieve during system analysis. Such analytical knowledge can be very powerful as it may accumulate over the years as experience and know-how, actually, as body of knowledge. Such aggregated analytical knowledge can come close to a theory, practically, it might be even way more useful. Nevertheless, it stays inductive in nature and comes with little or no system theoretic ambition and thinking.

See how enterprise architecture works. Enterprise architecture frameworks like TOGAF (The Open Group Architecture Framework) or the IBM Zachman Framework [63] are extremely useful as they accumulate terminology and, in a sense, know-how about the components of today's organizations, for which IT is always mission critical [23]. An enterprise architecture framework enables a jump start into the analysis of an enterprise architecture. An enterprise architecture framework goes beyond a modeling language, because it provides a proven method and seasoned structure for enterprise architecture description. Nevertheless, in order to create real value in the sense of a deep domain analysis [56], enterprise architecture must be teamed together with business analysis [40] efforts. Still, business analysis endeavors follow an engineering approach.

A system-theoretic model is as powerful as much as it can inspire and influence system developments and maybe also technological initiatives. In software engineering we find paradigms like object-orientation or agent-orientation [60]. These are technological paradigms. They are strong, because they encapsulate technological design patterns and best practices. Nevertheless, the metaphors they rely on are really just metaphors and no system theories. In the quasi-taxonomy of this paper, they do not belong to the realm of system theory as discussed in Sect. 2.1. Something similar can be said about computational models like the actor model [37] or process algebra [39,48]. These are reductionist models of phenomena of computing, concurrency in the aforementioned cases, however, no system theories in the sense of this Sect. 2.1.

3 Possible Pathways

3.1 Viable Software Engineering Life-Cycle

Large-scale systems enact very large software engineering projects. Very large software engineering projects are fundamentally different from even large software engineering projects. First, sometimes even the project character is entirely lost, because the system ever evolves and the project never stops. In very-large software engineering projects, management issues have to be treated as first class citizens. Without that it is very difficult to gain control over such projects. Very large projects must be made subject to business alignment and IT strategy. In extreme cases they need to be treated as part of corporate reengineering. This means that very large projects need organization [34]. Today's software processes deal with management issues; however, these are approached implicitly and therefore in a non-flexible manner. Each process provides an ad hoc solution to an arbitrary combination of management problems. However, what we are talking about here is beyond standard management. We are talking about projects that are so large that they show group dynamics [43] and need for cultural change management [59]. What is needed in future is software project management that is aware of the phenomena in these fields and knows exactly how to deal with them.

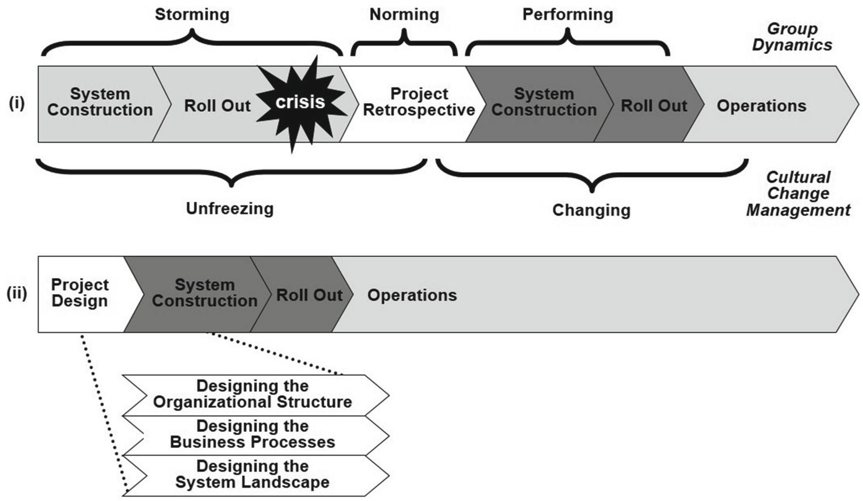


Fig. 2. Software engineering life-cycles.

In Fig. 2 we reconsider today's software engineering life cycle from a high-level, truly steering level. This means that the following discussion is not about concrete software processes. Rather, it is meant to apply to all of them. We consider a software project as consisting of three major stages, i.e., system construction, software roll out, and operation – see diagram (i) in Fig. 2. Now let us tell a story about how a typical project is often experienced. This is also shown in diagram (i) in Fig. 2. First, system construction needs more time than initially planned. Then, during roll-out the project runs into crisis. The users complain about the system. Project managers complain about the chief project management and the chief project manager complains about the development team, lack of IT strategy and the IT users. Now it has become necessary to clarify and settle responsibilities, to re-motivate or exclude troublemakers, and to re-convince users. A systematic way to do this is to enact a project retrospective in one form or another. In the sequel, a further, initially unplanned round of construction and roll-out must be started. As a consequence, a substantial cost and time overrun is encountered and still the system lacks behind initial expectations.

The problems that we have described here are called *storming* in group dynamics research [43] and *unfreezing* in organizational change research [59]. A viable software engineering life cycle [4] proactively manages storming and unfreezing. Therefore, it explicitly incorporates a project design phase as a first step, which can be seen in diagram (ii) in Fig. 2. Extra efforts are invested into project design to minimize risks. Initially, this costs extra, however, eventually it saves costs and time and increases quality. A major purpose of project design is to identify and address potential resistance as early as possible. The essence of the viable software engineering life cycle also shows in the emphasis on requirement

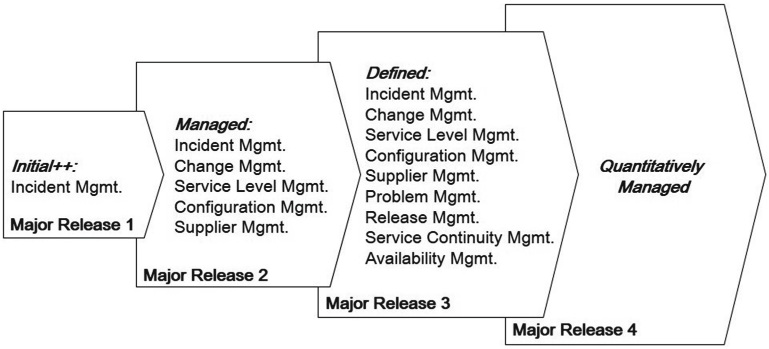


Fig. 3. Example viable software product.

elicitation. In very large projects, high-end engineering tasks must be fulfilled during requirement elicitation that can scale up to business process re-design or even corporate engineering [36], i.e., the restructuring of the organization. A viable software engineering life cycle will target the systematic incorporation of stakeholders, e.g., by a project clearing house or by a builders' hut. It might establish an employees' committee in addition to the standard steering committee. It will show inbuilt cultural change management and the early anticipation of software operations.

3.2 Viable Software Products

A viable software product [4] embodies cultural change [59]. Each major release of an enterprise IT system represents a maturity level. A viable software product supports several versions, each version representing a maturity level. Figure 3 shows an integrated IT service management platform as an example. The immediate introduction of a full-fledged product that contains all the features necessary for the highest achievable maturity is likely to fail. It would simply be too feature-rich and sophisticated to be introduced immediately. The cultural change caused by the introduction of a large software product should be handled in a step-by-step way. It should be realized using a pre-defined software version roadmap. The crucial point is the following. The creation of several versions in the viable software product portfolio is not merely about deactivating support for certain processes. A richer version is, in general, a non-conservative extension of the former version. This means that It does not merely add processes and features. Instead, it also changes the already supported processes and features. For example, the incident management in major release 2 in Fig. 3 might become significantly more complex when it operates in the context of problem management in major release 3. More complex means, e.g., more complex forms and reports, more options, more dialogue steps. A viable software product is a systematically evolving software product. The notion can be unified with the notion of software product line engineering [9].

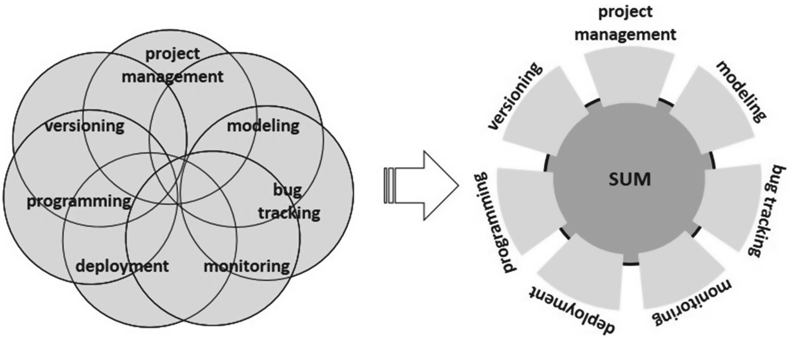


Fig. 4. Normalization and alignment of CASE tools.

3.3 Disruptive Software Engineering Platforms

In today's CASE tool landscapes we fight with artificial complexity, error-prone redundancies and poor tracebility of artifacts. In a view-based software development platform [4] all of the information about a software application is tightly integrated into a single underlying model (SUM) – see Fig. 4. The orthographic software modeling approach has been developed at the University of Mannheim [6–9]. In this approach, tools and their artifacts become a web of views onto this SUM. A software product is the collected information that exist about a software application and it is developed and maintained via its projections, i.e., via its views. Code plays no extra role. Code is also just a model and is tightly integrated into the SUM. Also, meta models [32,33,38], type systems [44] and constraints [16] are all unified in the SUM. A SUM represents the outcome of a deep standardization process of a software development domain. For each such domain, e.g., the domain of enterprise application development, all of its requirements and technological support is grasped and defined by such a deep standardization.

Deep standardization is not mere fiction. We have seen many successful instances of deep standardization in the past, e.g., the AS400 technology stack (OS400 / DB2 / TIMI /RPG), RAD (rapid development) tools, business process management suites. Albeit these are proofs of concept for deep standardization, there are proprietary and neither communicated nor exploited as deep standardizations. What is needed is to turn deep standardization into an open process. This is necessary due to the ever increasing complexity and speed of innovation cycles that we encounter in the domains that we are interested in. Concrete tools for streamlining the SUM are the maintenance of appropriate meta-information and a defined moderation process.

A particularly comprehensive and mature software engineering approach that is compatible with the view-based software development approach is the Living Models [14] approach. The Living Models approach also targets a tight and consistent integration of models. The viewpoints approach [30,52,58] is different from the view-based software development approach described here. The viewpoints approach started with a focus on requirements engineering [30]. Later, it was broadened to arbitrary artifacts [52]. Still then, it kept its focus on requirements. In particular, it stresses the problem of different stakeholders onto a software system. Viewpoints are distinct artifacts and not just projections of an underlying model [58]. They explicitly evolve separately from each other and they are also maintained separately. Then, knowledge is created by the construction of relationships between artifacts.

3.4 Next-Generation End-User Development

In our opinion, a key success factor for future engineering of large-scale system is in rising the abstraction level of development tools to the end-user level. We give three examples for such next-generation end-user development, i.e., typed business process specification, strictly typed enterprise portals and web weaving.

Typed Business Process Specification. Still, we encounter a gap between business process modeling and business automation [20], a gap between respective notations and technology. Mitigating these gaps has become even more important due to the ever increasing need and relevance of inter-organizational business processes in very large IT systems [31,51]. In form-oriented analysis [24–27] we model user dialogues in submit/response-style systems as typed, bipartite state machines that strictly alternate between client pages and server actions. This way we model the single user in a system dialogue in basic form-oriented analysis. In typed workflow charts [5], we refine server actions by the distinction of immediate server vs. deferred server actions to model worklists and workflows, coming up with a rigorous and executable specification mechanism for IT-based business processes – see Fig. 5. Current challenges are in the systematic decomposition of typed data flows and the integration of typed data flows with mainstream business process modeling notations.

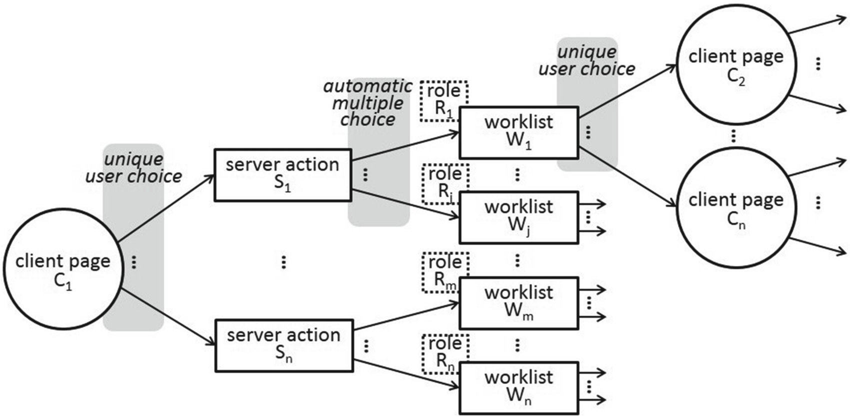


Fig. 5. Typed workflow chart.

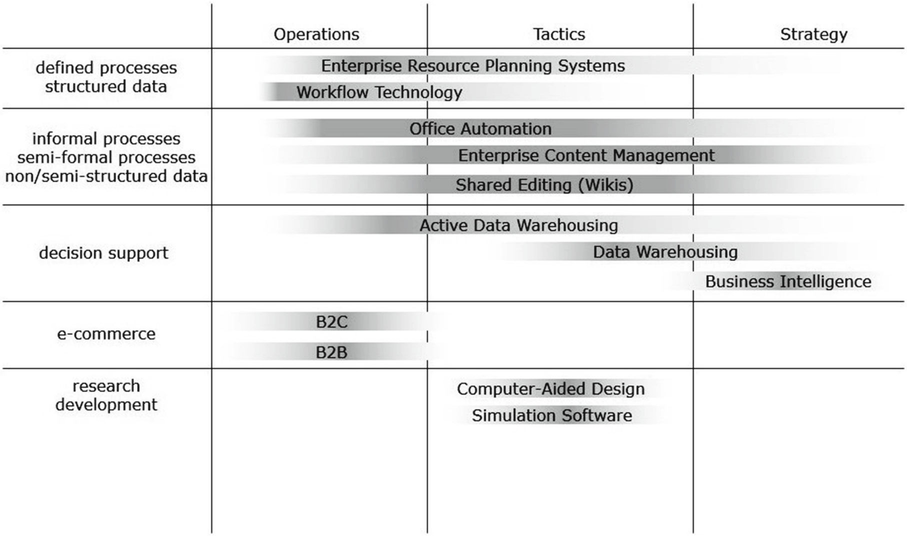


Fig. 6. Today's enterprise IT landscape.

Strictly Typed Enterprise Portals. The world of work changes and we will see ever more agile and flexible work methods and work organization in the future [18,19]. It is fitting that we currently experience a focus shift from business process management to knowledge management practices, a focus shift from analytical thinking to design thinking. Excellent IT support at all levels and in all fields, see Fig. 6, will be a crucial success factor in these transformations. But we need to rethink IT applications fundamentally. We see that process automation loses its dominant role in favor of other, more lightweight kinds of information systems. Enterprise content management, social software, Wikis, Web 2.0! Each of them stands for a special mix of features sometimes driven by a certain world view. We think that there is need to start from scratch. We should forget all biases and simply aim at understanding which IT application features are needed for what in an organization. How does the design of an ECM platform look like that is amenable to make enterprise content management a pervasive information system paradigm? What is needed is a balanced combination of access right control, versioning, collaborative data manipulation, transclusions [50] and data types.

Web Weaving. Web weaving [21] allows for weaving content and application hooks into existing web pages and applications - independent of ownership! The platform raises the Web 2.0 vision of ubiquitous web authorship to a next level of interactivity. Currently, new social software technologies gain ground in organizations. They come along with new agile and equal approaches to work organization. Classical enterprise resource planning (ERP) systems are also still growing in size and complexity. Therefore, still many organizations have their own soft-

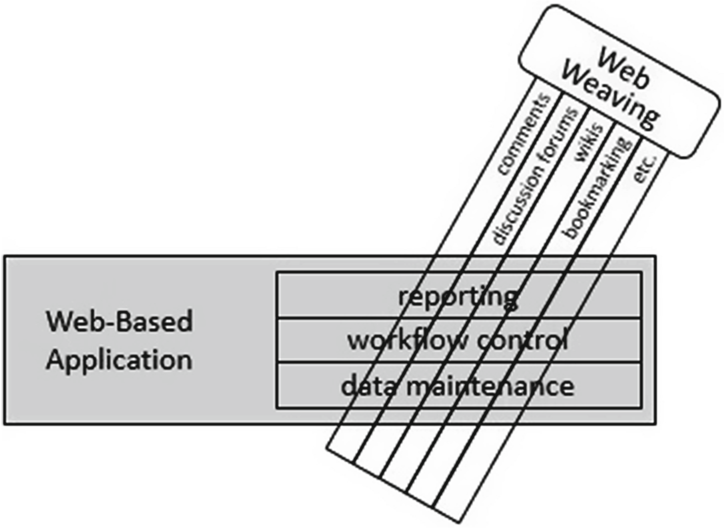


Fig. 7. Weaving content and application hooks into existing web-based applications.

ware development departments for realizing process-based applications. It seems that the demand for more and more process automation cannot be satisfied adequately. Computer-supported cooperative work (CSCW) [35] on the one hand and process-based automation [20] on the other hand co-exist in organizations with little to no integration. The same is even more true for the third strand of IT which is about individual office automation and individual ad-hoc IT support. With respect to this problem, web weaving offers a flexible and lightweight alternative to today's enterprise application integration approaches. Many important research issues arise with respect to web weaving: screen-to-page classification, preservation of consistency, refactoring issues, access right control, social grouping and tagging, security issues etc.

4 Conclusion

The information society and the ICT sector evolve together, in tandem. A stable, high-quality growth of the ICT sector is necessary for the sustainable development of the information society. Past initiatives to boost the ICT sector's productivity did not or did not yet take off to their full potential: open distributed systems, B2B, SOA, agility, offshoring, nearshoring, cloud computing, etc. A way out of the circle may be the more explicit connection of the engineering perspective with the system-theoretic and the normative perspective on IT-based systems. Against this background we have shared some thoughts on possible pathways to the future construction of large-scale systems: viable software engineering life-cycles, viable software products, disruptive software engineering platforms and next-generation end-user development (Fig. 7).

References

1. Alexander, C.: A Pattern Language - Towns, Buildings, Construction. Oxford University Press, Oxford (1977)
2. Alexander, C.: Patterns in Architecture. Keynote Speech, OOSPLA 1996 - Object-Oriented Programming, Systems, Languages, and Applications. Conference Video (1996)
3. Ashby, W.R.: Design for a Brain. Wiley, Hoboken (1954)
4. Atkinson, C., Draheim, D.: Cloud aided-software engineering: evolving viable software systems through a web of views. In: Mahmood, Z., Saeed, S. (eds.) Software Engineering Frameworks for Cloud Computing Paradigm, pp. 255–281. Springer, Heidelberg (2013)
5. Atkinson, C., Draheim, D., Geist, V.: Typed business process specification. In: Proceedings of the 14th IEEE International Enterprise Computing Conference, EDOC 2010. IEEE Press (2010)
6. Atkinson, C., Stoll, D., Tunjic, C.: Orthographic software modeling. In: Proceedings of the 2nd International Workshop on Models and Model-Driven Methods for Service Engineering, 3M4SE 2011. IEEE Press, August 2011
7. Atkinson, C., Stoll, D., Bostan, P.: Supporting view-based development through orthographic software modeling. In: Proceedings of the 4th International Conference on Evaluation on Novel Approaches to Software Engineering, ENASE 2009. INSTICC Press (2009)
8. Atkinson, C., Stoll, D.: Orthographic modeling environment. In: Fiadeiro, J.L., Inverardi, P. (eds.) FASE 2008. LNCS, vol. 4961, pp. 93–96. Springer, Heidelberg (2008)
9. Atkinson, C.: Component-Based Product Line Engineering with UML. Addison-Wesley, Boston (2002)
10. Beer, S.: Fanfare for Effective Management - Cybernetic Praxis in Government. The 3rd Richard Goodman Memorial Lecture, Delivered at Brighton Polytechnic, Moulsecoomb, Brighton, 14 February 1973
11. Beer, S.: The Heart of Enterprise - Companion Volume to: The Brain of the Firm. Wiley, Hoboken (1994)
12. Beer, S.: The Brain of the Firm - Companion Volume to: The Heart of Enterprise. Wiley, Hoboken (1994)
13. Bordbar, B., Draheim, D., Horn, M., Schulz, I., Weber, G.: Integrated model-based software development, data access, and data migration. In: Briand, L.C., Williams, C. (eds.) MoDELS 2005. LNCS, vol. 3713, pp. 382–396. Springer, Heidelberg (2005)
14. Breu, R., Agreiter, B., Farwick, M., Felderer, M., Hafner, M., Innerhofer-Oberperfler, F.: Living models - ten principles for change-driven software engineering. Int. J. Softw. Inform. **5**(1–2), 267–290 (2011)
15. Buxton, J.N., Randell, B.: Software Engineering - Report on a Conference Sponsored by the NATO Science Committee, Rome, October 1969. NATO Science Committee, April 1970
16. Draheim, D.: Reflective constraint writing. In: Hameurlain, A., et al. (eds.) TLDKS XXIV. LNCS, vol. 9510, pp. 1–60. Springer, Heidelberg (2016). doi:10.1007/978-3-662-49214-7_1
17. Draheim, D.: Towards total budgeting and the interactive budget warehouse. In: Piazolo, F., Felderer, M. (eds.) Innovation and Future of Enterprise Information Systems. LNISO, vol. 4, pp. 271–286. Springer, Heidelberg (2013)

18. Draheim, D.: Smart business process management. In: 2011 BPM and Workflow Handbook, Digital Edition. Workflow Management Coalition (2011)
19. Draheim, D.: The service-oriented metaphor deciphered. J. Comput. Sci. Eng. (2010)
20. Draheim, D.: Business Process Technology - A Unified View on Business Processes, Workflows and Enterprise Applications. Springer, Heidelberg (2010)
21. Draheim, D., Felderer, M., Pekar, V.: Weaving social software features into enterprise resource planning systems. In: Piazolo, F., Felderer, M. (eds.) Novel Methods and Technologies for Enterprise Information Systems. LNISO, vol. 8, pp. 223–237. Springer, Heidelberg (2014)
22. Draheim, D., Nathschlger, C.: A context-oriented synchronization approach. In: Electronic Proceedings of the 2nd International VLDB Workshop in Personalized Access, Profile Management, and Context Awareness, PersDB 2008 (2008)
23. Draheim, D., Weber, G. (eds.): Trends in Enterprise Application Architecture. LNCS, vol. 4473. Springer, Heidelberg (2007)
24. Draheim, D., Weber, G.: Form-Oriented Analysis - A New Methodology to Model Form-Based Applications. Springer, Heidelberg (2005)
25. Draheim, D., Weber, G.: Specification and generation of model 2 web interfaces. In: Masoodian, M., Jones, S., Rogers, B. (eds.) APCHI 2004. LNCS, vol. 3101, pp. 101–110. Springer, Heidelberg (2004)
26. Draheim, D., Weber, G.: Storyboarding form-based, interfaces. In: Proceedings of the 9th IFIP TC13 International Conference on Human-Computer Interaction, INTERACT 2003. IOS Press (2003)
27. Draheim, D., Weber, G.: Modeling submit/response style systems with form charts and dialogue constraints. In: Meersman, R. (ed.) OTM-WS 2003. LNCS, vol. 2889, pp. 267–278. Springer, Heidelberg (2003)
28. Draheim, D., Weber, G.: Strongly typed server pages. In: Halevy, A.Y., Gal, A. (eds.) NGITS 2002. LNCS, vol. 2382, p. 29. Springer, Heidelberg (2002)
29. The Ecomist. Blackrock - The Monolith and the Markets, 7th December 2013
30. Emmerich, W., Spanoudakis, G., Finkelstein, A.: Next-generation viewpoint-based environments. In: The 7th European Workshop on Next Generation of CASE Tools, NGCT 1996 (1996)
31. Eshuis, R., Norta, A.: A framework for service outsourcing using process views. In: Proceedings of the 14th International Enterprise Distributed Object Computing Conference, EDOC 2010. IEEE (2010)
32. Eessaar, E., Sgirka, R.: A SQL-database based meta-CASE system and its query subsystem. In: Sobh, T., Elleithy, K. (eds.) Innovations in Computing Sciences and Software Engineering, pp. 57–62. Springer, Heidelberg (2010)
33. Eessaar, E.: On pattern-based database design and implementation. In: Proceedings of the 6th International Conference on Software Engineering Research, Management and Applications, SERA 2008. IEEE (2008)
34. Gillette, W.: Managing megaprojects a focused approach. Software **13**(4) (1996). IEEE
35. Grudin, J.: Computer-supported cooperative work: history and focus. Computer **27**(5), 19–26 (1994). IEEE Press
36. Hammer, M., Champy, J.: Reengineering the Corporation: A Manifesto for Business Revolution. HarperCollins Publishers, New York (1993)
37. Hewitt, C., Bishop, P., Steiger, R.: A universal modular ACTOR formalism for artificial intelligence. In: Proceedings of the 3rd International Joint Conference on Artificial Intelligence, IJCAI 1973 (1973)

38. Himsl, M., Jabornig, D., Leithner, W., Regner, P., Wiesinger, T., Küng, J., Draheim, D.: An iterative process for adaptive meta- and instance modeling. In: Wagner, R., Revell, N., Pernul, G. (eds.) DEXA 2007. LNCS, vol. 4653, pp. 519–528. Springer, Heidelberg (2007)
39. Hoare, C.A.R.: Communicating Sequential Processes. Prentice-Hall, Upper Saddle River (1985)
40. IIBA. A Guide to the Business Analysis Body of Knowledge (BABOK Guide). International Institute of Business Analysis, Whitby (2015)
41. International Organization for Standardization. International Standard ISO 9000:2005(E). Quality Management Systems - Guidelines for Performance Improvements. ISO (2000)
42. Jensen, M., Meckling, W.: Theory of the firm - managerial behavior, agency costs, ownership structure. J. Financ. Econ. **3**(4) (1976)
43. Lewin, K., Conflicts, R.S.: Resolving Social Conflicts: Selected Papers on Group Dynamics. Harper & Row, New York (1948)
44. Lutteroth, C., Draheim, D., Weber, G.: Generative programming for C#. ACM SIGPLAN Not. **40**(8) (2005)
45. Maier, M.W.: Architecting principles for systems-of-systems. Syst. Eng. **1**(4), 267–284 (1998)
46. Mell, P., Grance, T.: The NIST Definition of Cloud Computing - version 15. National Institute of Standards and Technology, Information Technology Laboratory (2009)
47. Miller, E.: Designing Freedom, Regulating a Nation - Socialist Cybemetics in Allende's Chile. Working Paper #34, Massachusetts Institute of Technology, January 2002
48. Milner, R.: Communication and Concurrency. Prentice Hall, Upper Saddle River (1989)
49. Naur, P., Randell, B. (eds.): Software Engineering - Report on a Conference Sponsored by the NATO Science Committee, Garmisch, October 1968. NATO Science Committee, January 1969
50. Nelson, T.H.: The heart of connection - hypermedia unified by transclusion. Commun. ACM **38**(8) (1995)
51. Norta, A., Ma, L., Duan, Y., Rull, A., Klvart, M., Taveter, K.: eContractual choreography-language properties towards cross-organizational business collaboration. J. Internet Serv. Appl. **6**(8) (2014)
52. Nuseibeh, B., Kramer, J., Finkelstein, A.: Viewpoints: meaningful relationships are difficult! In: Proceedings of the 25th International Conference on Software Engineering, ICSE 2003, pp. 676–683. IEEE Press (2003)
53. Pappel, I., Pappel, I.: Methodology for measuring the digital capability of local governments. In: Proceedings of the 5th International Conference on Theory and Practice of Electronic Governance, ICEGOV 2011. ACM (2011)
54. Pappel, I., Pappel, I., Saarmann, M.: Digital records keeping to information governance in Estonian local governments. In: Proceedings of the International Conference on Information Society, i-Society 2012. IEEE (2011)
55. Piho, G., Tepandi, J., Parman, M.: Towards LIMS (laboratory information management systems) software in global context. In: Proceedings of the 35th International Convention of ICT, Electronics and Microelectronics, MIPRO 2012. IEEE (2012)
56. Piho, G., Tepandi, J., Roost, M.: Domain analysis with archetype patterns based Zachman Framework for enterprise architecture. In: Proceedings of the International Symposium on Information Technology. IEEE (2010)

57. Pollak, B. (ed.): Ultra-Large-Scale Systems - The Software Challenge of the Future. Software Engineering Institute, Carnegie-Mellon University, Pittsburgh (2006)
58. Sabetzadeh, M., Finkelstein, A., Goedicke, M.: Viewpoints. In: Encyclopedia of Software Engineering. Taylor & Francis, pp. 1318–1329 (2010)
59. Schein, E.H.: Organizational Culture and Leadership. Wiley, Hoboken (2004)
60. Weiss, G. (ed.): Multiagent Systems, 2nd edn. MIT Press, Cambridge (2013)
61. Wiener, N.: The Human Use of Human Beings. Houghton Mifflin, Boston (1950)
62. Wiener, N.: Cybernetics - or Control and Communication in the Animal and the Machine. Wiley, New York (1948)
63. Zachman, J.A.: A framework for information systems architecture. IBM Syst. J. **26**(3) (1987)

An Overview of Digital Signing and the Influencing Factors in Estonian Local Governments

Sigrid Felt[1], Ingmar Pappel[2], and Ingrid Pappel[2](✉)

[1] Interinx Ltd., Tallinn, Estonia
sigrid.felt@interinx.com

[2] Large-Scale Systems Group, Technical University of Tallinn, Tallinn, Estonia
{ingmar.pappel,ingrid.pappel}@ttu.ee

Abstract. In Estonia, digital signing was started with Digital Signatures Act already in 2000, so that for years, the priority of the state has been to make digital signing and its use with various types of documents more efficient. This article provides a study of the use of digital signatures with documents related to decision-making processes and analyses the factors which influence this. Local governments have been used as an example to provide an overview of the digital signing statistics for local government document exchange. The article highlights the differences related to the size and administrative capacity of the local governments as well as their readiness to transition into the information society.

Keywords: Digital signing · Digital document exchange · Digital administration

1 Introduction

We live in an increasingly digitalized world. In addition to the different technological solutions in everyday life, document management and the related decision-making processes have also become digital. The Digital Agenda for Estonia 2020 aims for a "simpler state 2020" [1], whereby in order to make the public sector more effective, it is important to achieve a 95 % paperless official communication rate by 2020. This requires local government services to be as electronic as possible and that as an end result of the provided services, instead of printing out a paper to prove the fact of service provision, it is stored in digital form. In order to achieve this, various procedural systems are in use in Estonia, including document management systems (DMS) – which comprises of and manages documents as well as facilitates constant access to them. DMS has brought transparency to administration and allowed for including citizens in the decision-making processes of the organization. This, in turn, has made the implementation of digital signatures more efficient in Estonia.

In this paper the correlation between the use of digital signatures and specific document types is discussed based on usage of the DMS Amphora. Additionally, a survey has been conducted that provides an overview of the factors influencing digital signing in local governments. Various research methods were used to carry out this

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 371–384, 2016.
DOI: 10.1007/978-3-319-48057-2_26

survey, such as data obtained from databases on the basis of specified criteria, the observation of world practices, questionnaires and interviews. Generalizations have been made based on more than 50 % of the Estonian local governments.

In order to make digital document exchange more efficient, several solutions have been developed in Estonia [2], e.g. the document exchange center (DEC) and e-services at the citizen portal eesti.ee environment, which enable the digital processing and management of a document life cycle from its birth to death. Over the years, the volume of paper documents exchanged between authorities has decreased significantly [3], which in turn has a positive effect on the budget of the institution. DMS Amphora is used in 127 Estonian local governments and this article presents the data from 117 Estonian local governments because their data was available in the database in the proper form. The data has been taken about the first quarter of 2016. The software solution enables to observe the reply deadline for the letters, and to digitally sign all documents and letters. The data used in this work have been obtained with SQL queries from the DMS database according to the following:

- How many incoming documents has the given local government registered in the Amphora document management system;
- How many outgoing documents has the given local government registered
- How many of the outgoing documents has the given local government signed digitally;
- Total numbers of letters and documents;
- Capability index of the local government units [4];
- Capability ranking of the local government units;
- Number of residents in the given local government;
- How many documents per residents are there in the document management systems in the first quarter;
- Name of the local government;
- County in which the local government is located;

In Sect. 2 we explain the background, i.e., the current state of digital signing in Estonia and its motivation. Also we report on first insight concerning problems with digital signing and digital archiving. In Sect. 3 we provide the results of a survey concerning digital signing. In Sects. 5 and 6 we derive factors resp. recommendations for digital signing from the survey results. We discuss related work in Sect. 6 and finish the paper with a conclusion in Sect. 7.

2 Digital Signatures in Estonia

As aforementioned before, the digital signatures in Estonia is governed by the Digital Signatures Act (DAS), which was adopted on 7 March 2000 [5]. In the eyes of the law, a digital signature is equal to a handwritten signature. All Estonian authorities are required to accept digitally signed documents. Estonian public authorities are required to accept digitally signed documents. Two certificates are issued along with an ID-card. One certificate is for identification and the other for digital signatures. It is important to ensure that these certificates have not expired when using digital signatures, and it is also essential to know PIN1 and PIN2. In addition to signing using the ID-card, Mobile-ID signatures are becoming increasingly popular. In 2015, the number of Mobile-ID users increased by 40 %, exceeding the 75,000 user line this January. These users carried out over 25 million Mobile-ID transactions in the last year. If in 2014, Mobile-ID was used for an average of 1.8 million transactions per month, then last year the monthly average was 2.7 million [6]. Three types of formats are used in Estonia – BDOC, DDOC, and CDOC [7]. The oldest one of these, the original is the DDOC. BDOC is a newer format meant for replacing the DDOC format, and it is certainly more consistent with international standards. CDOC is a file which in its encrypted form contains a data file (XML document or other binary file, e.g. MS Word, Excel, PDF, RTF, etc.), the certificate of the recipient, an encrypted key for data file decryption, and other optional metadata [8].

2.1 Reasons for Using Digital Signatures

A digital signature is the counterpart of an ordinary signature used to sign information in digital form. Digital signatures help identify the link between the document and the person who signed it. A digital signature along with a time stamp forms a combined dataset with the document, the components of which cannot be individually altered at a later time. Digital signatures replace ordinary signatures which helps to ensure the authenticity and security of electronic documents. Besides apply paperless administration to enable digital document exchange [9]. Ensuring security with a digital signature means that the document author is known and the document has not been altered by third parties between being sent and received [10]. The digital signature standard (DSS) was created by the US National Security Agency. DSS is based on the digital signature algorithm (DSA). DSS can only be used for digital signatures but the DSA can also be employed for encryption [11]. The simplicity of digital signing can be considered its biggest advantage. It is quick and convenient and lacks many of the risks that signing on paper entails. It is certain that a physical person is responsible for the signature. The signed document has not been subsequently edited by third parties, this option is eliminated by mathematical links. It is always possible the check the signing date because the time stamp is a part of digital signing.

- An endless number of legally equal copies can be made of a digitally signed document.
- Digital documents do not take up physical space.
- Digital documents do not require paper, a printer or other superfluous resources.
- Digital documents do not need to be delivered and communication is possible through electronic channels.
- With the use of DMS, digital documents can be found more quickly and archived on the basis of very different criteria.

When signing digitally, one must consider that the generated file can be singly read using convenient methods by all interested parties and that it can be opened without issues in the future as well. If a file has been signed in one format, then it cannot be converted into another format without losing the signature. It is important to use to correct file formats for signing so that the file meets all the requirements. There are several possible purposes for using digital signatures, e.g. no need to specifically meet in person for a signature or to send documents with ordinary mail, thus significantly saving time. Digital signing allows for automating activities and to reduce spending time on regularly signing a large number of documents physically. If necessary, the document should be encrypted so strangers cannot read it.

2.2 Problems Related to Digital Signing

From a local government perspective, several issues have been highlighted that are related both to the organizational as well as technical aspects. Also, this is widely discussed elsewhere as well [12]. From a technical point of view, the digital signature format can be limited, as it is possible that different environments can show the document in different ways. The most important and serious risk with using digital signatures is that the signature rights can be stolen with a private key – the owner of the certificate must carefully monitor that the private key does not leave the possession of the signature owner. Nowadays, different methods have been devised to tackle this and the risk is diminishing.

The problems that may arise when using digital documents tend to differ between small- and large-scale uses. In both cases, one must bear in mind that not all clients and partners may have an ID-card or Mobile-ID and that parallel paper document use must be retained. The latter can only be avoided when an authority issues unilaterally signed documents. This could create duplication. For small-scale use, e.g. internal use of an organization and signing contracts with larger partners and clients, different issues occur and the use of a computer and ID-card and passwords is an extra effort, takes more time and is not suitable in outdoor conditions. In addition, a problem with digital documents may arise regarding the accompanying time stamp – the physical time of signing is visible to everyone who looks at the document. In local governments, this is linked to certain decisions and the granting of rights, where an important administrative act is formalized after the fact, so to speak.

2.3 Problems Related to Archiving

Many local governments have brought out archiving as an issue for digital signing. Archiving digitally signed documents requires some extra effort [13, 14]. With archiving, one must take into account that in addition to digital documents, paper documents also need to be managed. Thus, hybrid files are created. Inevitably, it is more difficult to use two separate management systems rather than only have one; it is reasonable to manage both digital and paper documents in the same information system. A solution is that the location, existence, and main information (what type of document, what parties, when, etc.) about the paper documents is registered in the same information system and in the same way as for digital documents, in the simplest case by using a small ordinary document file containing the main information. If an organization already employs a paper document registration system, adding digital document management to the same system is likely to be the most effective – provided that this is technically possible.

Regarding potential software solutions, it is important to consider whether an existing software already in use could be suitable for archiving digital documents, or if the standard activities used in the organization already could not be employed for archiving the files. For instance, digital documents could simply be stored in a file system, grouping them chronologically into year- or month-based catalogues and coding the critical information (client name or code) into the file name. This can be used if there is a relatively small amount of digital documents. In addition, an existing specialized archiving software could be used and generally, a DMS already contains an archiving function. If it does not, a suitable archiving software could be created for the organization.

2.4 Digital Signatures and Digital Document Authenticity

Is a digital signature always a sufficient guarantee of the digital document authenticity for digital archiving? From the perspective of the Estonian national archive, it can be not sufficient [15–17]. A digital signature does protect the signed information (the content of the document) from unwanted changes but it is not enough to completely understand the document. A part of the information no less important than the content is hidden in the links between the documents – these allow us to understand the activities of the organization, during which the document was created. A digital signature does not release an organization from good and controlled management of the document, which is one of the guarantees of document authenticity. In the case of signed, but especially for digital documents with a permanent retention period, the organization must implement and ensure specific policies and procedures that enable verifying the creation, sending, forwarding, retention, and separation of documents [16].

In the future, it is possible to use archival time-stamping for ensuring the long-term preservation of documents in the BDOC format. This mechanism is based on the principle “fortify that, which could be weak” [18]. Consecutive time stamps protect the entire contents from weak hash algorithms and from breaching cryptographic material and algorithms. Certain costs are associated with this, as there is a need to enter into a contracts with an organization that offers certification and time stamping services (presently, in Estonia, this organization is Certification Centre). Monthly bills also need to be paid for the validity confirmation service, however, the costs are not that big.

3 Digital Signature Statistics Based on DMS Databases

The study presented in this article only reflects the digital signing of documents exchanged using the DMS, but many documents are processed outside of the document management system using other components [19]. For instance, if one were to change one’s place of residence and make an application about this to the local government, this application is registered as an entry in the Population Register and may well not be reflected in the document management system. The same applies to construction permits, authorizations for use, and applications for design criteria, which are all registered in the Construction Register. The data that are registered in the social services and benefits data register (STAR) are also excluded from the document management system. In Estonia, information exchange with other systems is mainly carried out over the X-road for relational systems [20]. However, this is not always the case, and therefore it is necessary to also observe the situations where information with external systems is exchanged outside the X-road, in order to have adequate statistics about the public sector document exchange. Although X-road is the preferred communication channel, there are still information systems that communicate directly, i.e. exchange documents by other interfaces. Below, data is shown in various groups (local government totals, more successful local governments, less successful local governments, etc.), bringing out volume of digitally signed documents (Fig. 1) (Table 1, 2, 3, 4 and 5).

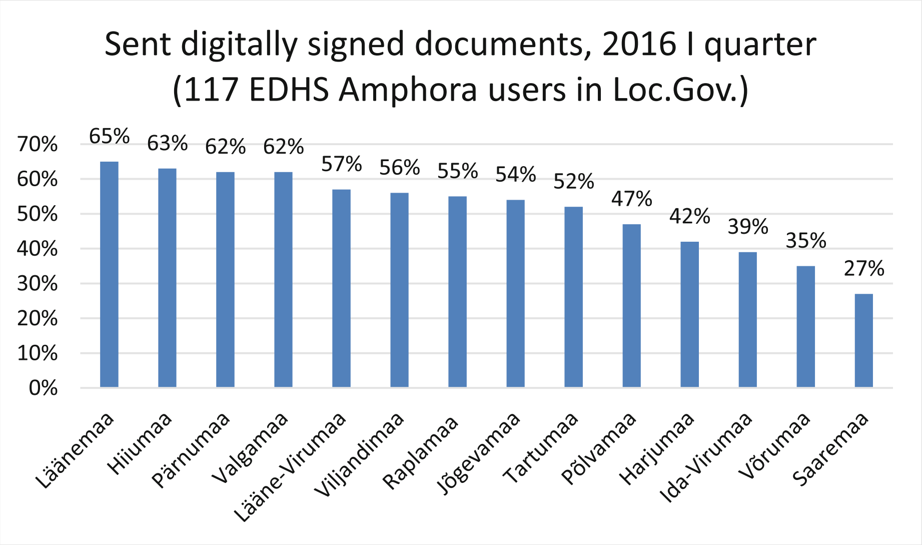


Fig. 1. Summarization according to counties

Table 1. Consolidated data

Consolidated data were as follows:	
Total number of sent documents	24801
Total number of digitally signed sent documents	12245
Percentage of digital signing for sent documents	49 %
Number of local governments in the sample	117
Average number of residents in local governments in the sample	3298

Table 2. Local governments that use digital signing the most

Local government	County	Sent digitally signed documents	Number of residents	Local government capability index ranking
Tori parish	Pärnu	86 %	2327	127
Elva town	Tartu	85 %	5768	39
Värska parish	Põlva	81 %	1374	113
Tahkuranna parish	Pärnu	81 %	2389	114
Audru parish	Pärnu	81 %	5858	52
Karksi parish	Viljandi	80 %	3400	88
Vigala parish	Raplamaa	79 %	1267	66
Paikuse parish	Pärnumaa	75 %	3899	74
Kehtna parish	Raplamaa	75 %	4459	49
Vinni parish	Lääne-Viru county	75 %	4757	21

Table 3. Local governments that use digital signing the least

Local government	County	Sent digitally signed documents	Number of residents	Local government capability index ranking
Pihtla parish	Saare county	2 %	1411	109
Ahja parish	Põlva county	0 %	1011	191
Kihelkonna parish	Saare county	0 %	773	86
Laimjala parish	Saare county	0 %	711	183
Meeksi parish	Tartu county	0 %	594	194
Mustjala parish	Saare county	0 %	691	207
Sõmerpalu parish	Võru county	0 %	1799	118
Torgu parish	Saare county	0 %	350	208
Torma parish	Jõgeva county	0 %	1991	137
Varstu parish	Võru county	0 %	1075	180

Table 4. Most digitally signed letters per resident in local governments with up to 10,000 residents

Local government	County	Sent digitally signed documents per resident	Number of residents	Local government capability index ranking
Lüganuse parish	Ida-Viru county	0.235	3014	23
Vihula parish	Lääne-Viru county	0.117	1955	36
Piirissaare parish	Tartu county	0.098	102	210
Vormsi parish	Lääne county	0.096	415	75
Misso parish	Võru county	0.096	645	126
Meremäe parish	Võru county	0.092	1093	181
Mõniste parish	Võru county	0.084	873	166
Kernu parish	Harju county	0.084	2040	27
Värska parish	Põlva county	0.080	1374	113
Are parish	Pärnu county	0.069	1297	122

Table 5. Number of digitally signed letters per resident in local governments with more than 10,000 residents

Local government	County	Sent digitally signed documents per resident	Number of residents	Local government capability index ranking
Viimsi parish	Harju county	0.010	18430	4
Viljandi town	Viljandi county	0.028	18111	32
Rae parish	Harju county	0.035	15966	1
Rakvere town	Lääne-Viru county	0.021	15942	40
Maardu town	Harju county	0.005	15676	29
Saue parish	Harju county	0.032	10451	7
Haapsalu town	Lääne county	0.037	10425	41

4 Factors Influencing Digital Signing

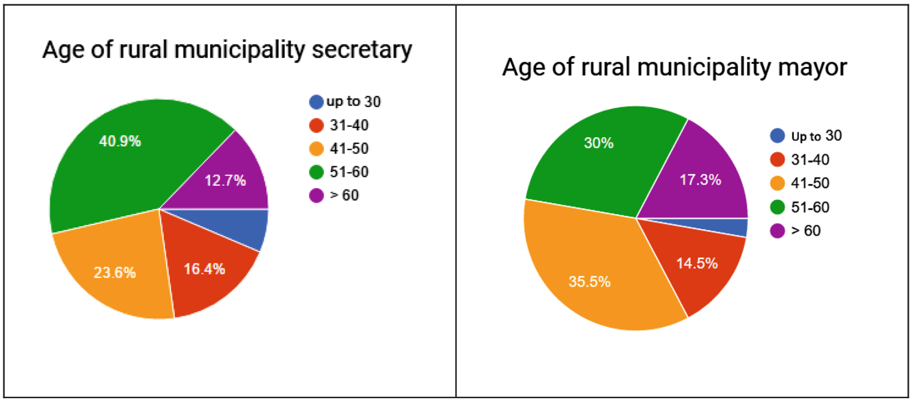
In this section we delve into the factors influencing digital signing by seeking for generalizations based on survey. This analysis is based on the survey conducted in spring 2016, which examined the various factors that influence the implementation of digital signing in local governments. The answers obtained from the survey illustrate the main factors which obstruct or advance digital signing in DMS. The answers reflects different criteria and measurements sets concerning the digital signing. For instance, answers to the questions "Do you sign government legislation digitally" the "yes" was answered 39.3 %. Question "Do you sign outgoing documents digitally" got 58.2 % "yes" and "partially" 40 %. "Do you think preserving digital signatures is safe?" gave 46.4 % "maybe" and 49,1 "yes". Question "Do you think digital signatures can be used as evidence (e.g. in court)" got 79.8 % "yes" answers. To the question "Is forwarding digitally signed documents to citizens an issue" gave 57.4 % of "Yes answers". On the following figures are shown different criteria which were investigated such us variety of age and different factors influencing the digital signing (Figs. 2, 3 and 4).

Also, in the inquiry, there was an open text question "What should be done to introduce the digital signing in depth". The most used suggestions were brought out as follows:

- In order to raise elder people capability, the access to a computer, internet should be guaranteed more widely
- Digital signing should be introduced (forced) by rural municipality mayor within organisation (local government)
- Raise awareness regarding the digital archiving – explain long-term preservation methods
- It is necessary introduce and market digital signing for both - officials and citizens
- Develop more Public Internet Access points (for instance use county's library), which gives the opportunity to consume public e-services (different application)

Allover, from the survey, we learned that the following are the delaminating factors for digital signing:

- *Digital Divide*
  - elder people vs. younger people
  - lack of ubiquities internet access
- *Lack of sponsorship*. Lack of sponsorship by leaders in administrations.
- *Lack of awareness concerning digital archiving.*
- *Lack of iniquitousness towards population.* Barrier in the usage of digital signing between officials and citizens.



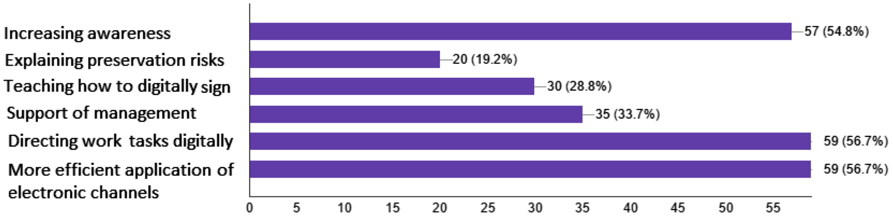


Fig. 2. How to raise digital signing?

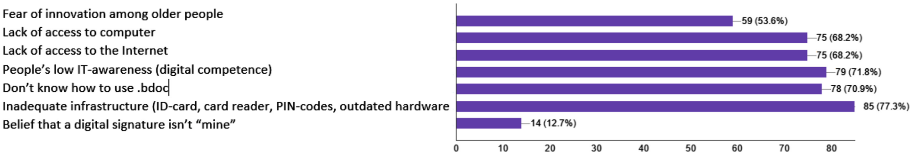


Fig. 3. Please mark the factors which could prevent forwarding digitally signed documents to citizens

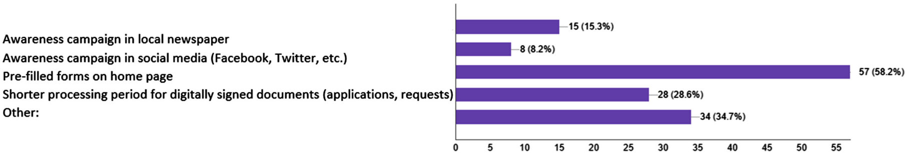


Fig. 4. How do you motivate your citizens to use digital signatures?

5 Recommendations for Implementing Digital Signatures

In order to implement the digital signing of documents more efficiently, it is necessary to consider the most suitable scenarios regarding saving/archiving documents. In most cases, there is no need to develop or implement some kind of special software, and using freely available standard software will suffice. Implementing digital signatures in the DMS requires certain changes which can be divided into organizational and technological. On the information technology level, the work of users should be made convenient and where possible, automatic storage of deliveries and work task flows should be introduced, as these support digital signing. On the organizational level, potential activities are mainly linked with training and increasing user awareness, both regarding simplifying work flows and digital archiving. If automatic work flow simplification is not possible, awareness campaigns are required and the users need to be taught how to a document is sent to be digitally signed when registered in the DMS or forwarded from the DMS. Digital signing is closely related to the implementation of digital records management. If the work processes are digital, digital signing is one logical step in the whole process. When discussing the digitalization of processes, it is important to note the complexity of business processes which in many cases are related to the size of the organization and the complexity of the offered services. For organizations with a rather large number of users, the complexity and large amount of business processes will be the deciding factors, nevertheless the people signing digitally tend to be the leaders of the organizations.

It is important to consider mapping, selecting, and analyzing the business processes suitable for the paperless alternative. The work load ranges from a few days to half a year, depending on the organization and the complexity of the task. Vitally, the preparation and carrying out of archiving digital documents must be planned. If the organization already has the required software or experience of using ordinary software, developing the principles for digital archiving is going to be easier.

Transitioning to digital signing in Estonia is also supported by a European Commission directive called eIDAS [21] that enters into force on 1 July 2017. The standards listed in this directive also include the bdoc-format digital signature used in Estonia. European public authorities are required to recognize digital signatures that meet this standard, thus providing an Estonian citizen with the right to bring an action against someone in a court in Barcelona that is signed digitally. On the other hand, Estonian public authorities have to learn to receive other types of digital signatures received from Europe. Estonian digital signatures must start accepting digitally signed documents with an equal or "stronger" signature from other European Union countries. Estonian citizens in turn get the opportunity to turn to other European public authorities with their digitally signed documents.

6 Related Works

Digital signing Problems related to digital signing are widely discussed from the perspective of the integrity and authenticity [12], and digitally signed documents requires extra effort for digital archiving [13–15]. In order to guarantee the organization in Estonia must implement and ensure specific policies and procedures [17], besides the initiative comes from the EU level as well [22]. However, investigating other countries experiences several circumstances indicates the rise of the digital signing. Levy [24] recognizes that *"to benefit from its massive advantages, digital signatures still have challenges to overcome"*.

According to Little´s report [24] the financial services industry has been the pioneer in the adoption and development of digital signature solutions, and he expects other industries, such as telecommunications, commerce, utilities, notaries and healthcare, to follow suit. Estonian case shows that besides the financial service industries the public sector has been adopted digital signatures quite well as well. However, based on Little´ s report´s the findings [24] are claiming that *"challenges include the integration and alignment of the technology with existing processes, together with a transparent analysis of the related regulatory situation and its legal consequences when implementing digital signatures"* On this basis, it should be admitted that same matter must be considered in Estonian case. Although, the digital signatures are more efficient way to work, still the different obstacles should be resolved first. Besides the legal framework, the problems related to digital signing are tight to technology issues and people's resistance. This is discussed in the study conducted in USA where survey [23] shows *that "digital signatures have emerged as one of the technology priorities for local and state governments for the purpose of gaining both operational efficiencies and legal assurances"*. Like to this paper, the aforementioned survey was conducted among the local and state authorities and shows many similarities in findings to this work here as well. Still, the main advantages of this presented work are presenting besides the qualitative research results based on statistics from the DMS databases. This in turn gives real-live numbers of the actual signing of the local governments and qualitative research helps to understand the difference of the curve within local governments. To conclude, the international studies are indicating that digital signing is an important future trend and its development should be considered, while making local governments work routines more efficient along with the cost savings on paper products.

7 Conclusion

Digital signing has already claimed a significant place in today's society but signs are showing that the importance of digital signing is bound to increase even more in the near future. Firstly, the simplicity and security of the signature make it a preferred choice ahead of signing on paper. Secondly, digital document exchange also translates into savings in the budget. It can also help increase the security of the documents: a digital signature is tamper-proof and creates the option of creating an unlimited number of authentic verifiable copies of the document. This in turn enables to reduce the work load and increase the efficiency of local governments. Although the survey revealed that many of the smaller local governments do not have such administrative capabilities, the proportion of digital signatures is still notable. The main findings of the survey can be summarized as limiting factors concerning digital signing, which are digital divide, lack of sponsorship, lack of awareness concerning digital archiving and lack of iniquitousness towards population. For more efficient implementation, in addition to technological adaptations, the awareness of officials about issues related to digital archiving as well as software capabilities and interoperability for reading documents should be increased.

References

1. Ministry of Economic Affairs and Communications. Digital Agenda 2020 for Estonia. Ministry of Economic Affairs and Communications (2013)
2. Pappel, I., Pappel, I.: Implementation of service-based e-government and establishment of state IT components interoperability at local authorities. In: The 3rd IEEE International Conference on Advanced Computer Control (ICACC 2011), Harbin, Hiina, 18–20 January 2011, pp. 371−378. Institute of Electronics and Computer Science, Singapore (2011)
3. Outcome of the 2014 Estonian document exchange classification Project DECS
4. Classification of Estonian administrative units and settlements 2015v1
5. Digital Signatures Act. https://www.riigiteataja.ee/en/eli/508072014007/consolide
6. (2000). http://www.id.ee/?id=30009&read=37530
7. http://pcsupport.about.com/od/fileextensions/f/ddocfile.htm
8. Digidoc file formats (2014)
9. Pappel, I., Pappel, I., Saarmann, M.: Digital records keeping to information governance in Estonian local governments. In: Shoniregun, C.A., Akmayeva, G.A. (ed.) i-Society Proceedings: i-Society 2012, 25–28 June 2012, London, UK, pp. 199−204. IEEE, London (2012)
10. Merkle, R.C.: A certified digital signature. In: Brassard, G. (ed.) Advances in Cryptology — CRYPTO 1989, LNCS, vol. 435, pp. 218–238. Springer, Heidelberg (2011). http://link.springer.com/chapter/10.1007%2F0-387-34805-0_21
11. Naccache, D., M'RaÏhi, D., Vaudenay, S., Raphaeli, D.: Can D.S.A. be improved? — Complexity trade-offs with the digital signature standard. In: De Santis, A. (ed.) Advances in Cryptology — EUROCRYPT 1994, LNCS, vol. 950, pp 77–85 (2006). http://link.springer.com/chapter/10.1007/BFb0053426
12. Vigila, M., Buchmanna, J., Cabarcasb, D., Weinerta, C., Wiesmaierc, A.: Integrity, authenticity, non-repudiation, and proof of existence for long-term archiving: a survey. Comput. Secur. **50**, 16–32 (2015). Elsevier http://www.sciencedirect.com/science/article/pii/S0167404814001849
13. Wallace, C., Pordesch, U., Brandner, R.: Long-term archive service requirements (2007). http://www.rfc-editor.org/info/rfc4810
14. Lekkasa, D., Dimitris Gritzalisb, D.: Long-term verifiability of the electronic healthcare records' authenticity. Int. J. Med. Inform. **76**(5–6) 442–448 (2007). http://www.sciencedirect.com/science/article/pii/S1386505606002152
15. Lynch, C.: The future of personal digital archiving: defining the research agendas. In: Hawkins, D.T. (ed.) Personal Archiving: Preserving Our Digital Heritage. Information Today (2013). http://stage.cni.org/wp-content/uploads/2013/09/Personal-Digital-Archiving-Cliff-Lynch-Oct-29-2013.pdf
16. The 2005-2010 National Archives' digital archives strategy. http://www.arhiiv.ee/public/Digiarhiiv/digistrateegia.pdf
17. Digital archives vision (2005). http://www.arhiiv.ee/public/Digiarhiiv/da_visioon.pdf
18. Archives management requirements for digital records (2008). http://www.arhiiv.ee/public/Juhised/digidok_arhiveerimine.pdf
19. http://www.arhiiv.ee/et/dokumendid-rahvusarhiivis-ja-digitaalallkiri/&i=

20. Kalja, A., Robal, T., Vallner, U.: New generations of Estonian eGovernment components. In: Portland International Conference on Management of Engineering and Technology (PICMET) (2015). http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=7273002&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D7273002
21. Draheim, D., Koosapoeg, K., Lauk, M., Pappel, I., Pappel, I., Tepandi, J.: The design of the Estonian governmental document exchange classification framework. In: Kő, A., Francesconi, E. (eds.) EGOVIS 2016. LNCS, vol. 9831, pp. 33–47. Springer, Heidelberg (2016)
22. https://ec.europa.eu/digital-single-market/en/trust-services-and-eid
23. American City & Council Study Findings. Benchmark Survey: Digital Signatures. ARX (2014)
24. Levy, D., Schaettgen, N., Duvaud-Schelnast, J., Socol, S.: Digital Signatures - Paving the Way to a Digital Europe. Arthur D. Little, Boston (2014)

Aggregating Service Level Agreements in Services Bundling: A Semiring-Based Approach

Trung-Viet Nguyen[1,2], Lam-Son Lê[1(✉)], and Khuong Nguyen-An[1]

[1] Faculty of Computer Science and Engineering,
HCMC University of Technology, Ho Chi Minh City, Vietnam
lam-son.le@alumni.epfl.ch, nakhuong@hcmut.edu.vn

[2] Faculty of Information Technology, Can Tho University of Technology,
Can Tho City, Vietnam
ntviet@ctuet.edu.vn

Abstract. Business services arguably play a central role in service-based information systems as they fill in the gap between the technicality of Service-Oriented Architecture and the business aspects captured in Enterprise Architecture. Business services have distinctive features that are not typically observed in Web services, e.g. significant portions of the functionality of business services might be executed in a human-mediated fashion. As such, service level agreement (SLA) should be described as a mixture of human-mediated functionality (e.g., service penalty) and computer-interpretable measurement (e.g., reliability, payment). In this paper, we propose a formal framework for reasoning about the SLAs from the perspective of services bundling – the practice of innovatively organizing business services into a bulkier service offering that creates new values. Specifically, we (a) represent multi-level SLA of a business service in terms of service reliability, payment and penalty using the mathematical structure of semiring; (b) provide formality for aggregating SLAs of the constituent services that make up the service bundling; (c) make multi-level SLAs of a bundled service technically comparable. The main contribution of this work is a machinery for handling a large number of SLAs generated through services bundling, allowing to the service consumers to pick up the right service offering according to their preference.

Keywords: SLA · Services bundling · Semiring · Formal methods

1 Introduction

In the last few years, service-oriented computing has become an emerging research topic in response to the shift from product-oriented economy to service-oriented economy. On the one hand, we now live in a growing services-based economy in which every product today has virtually a service component to it [21]. In this context, services are increasingly provided in different ways in order to meet growing customer demands. Business domains involving large and complex

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 385–399, 2016.
DOI: 10.1007/978-3-319-48057-2_27

collection of loosely coupled services provided by autonomous enterprises are becoming increasingly prevalent [1,24]. On the other hand, Information Technology (IT) has now been thoroughly integrated into our daily life [14] and gradually gives rise to the paradigm of ubiquitous computing. As such, business services are essentially IT-enabled making the border between business services[1] and IT-enabled services blurred. At the high-level operationalization of a business service, we see business activities happening between service stakeholders. We may or may not witness IT operations at this representational level. At lower levels, the operationalization of these services are eventually translated into IT operations as we have seen in the cases of banking services, recruitment services, library services, auctioning services, etc.

Services bundling is a practice of innovatively grouping related business services to come up with new service offerings that create new service values for customers. A typical example of service bundling is car rental, accommodation, travel insurance could be combined to offer a valued travel package to tourists. The customers may experience to-be-bundled business services in two different ways: by consuming them individually and by taking them via service bundling. As such, aggregating service level agreement (SLA) of business services when bundling them poses a number of challenging questions. First, the SLA of business services are to be perceived from at least two different angles, specifically the customer's point of view and through the provider of the service bundling (e.g. travel agency). Second, as the SLA of a service might have multiple levels (hence, the term multi-level SLA), bundling services could results in generating a large number of SLAs that are to be perceived by the customers. We are in need of a machinery that sorts them and helps the customers choose the right SLA according to their preference. While there exists considerable amount of work on reasoning about SLA of Web services, not much effort has been put in coping with the complexity of the SLAs from the business standpoint. In this paper, we propose a formal framework for reasoning about the SLAs for service bundling. Specifically, we (a) represent multi-level SLA of a business service in terms of service reliability, payment and penalty using the mathematical structure of semiring; (b) provide formality for aggregating SLAs of the constituent services that make up the service bundling; (c) make multi-level SLAs of a bundled service technically comparable. This work sheds light on how human-mediated concepts such as business contracts [12,17] could be reasoned about together with computer-interpretable aspects such as costs and reliability in SLA.

The remaining of this paper is organized as follows. In Sect. 2, we give preliminaries for SLA, service penalty and the semiring. Section 3 expresses our research approach and describes a running example. Section 4 is the core of the paper – we formally define multi-level SLAs and reason about aggregating them

[1] By calling them business services, we mean services happening between people or business entities. They are enabled by IT in one way or another. For the sake of simplicity, we shall use the term "business service" or simply "service" to refer to these IT-enabled business services throughout this paper.

in services bundling. We survey related work in Sect. 5. Section 6 ends the paper by drawing some concluding remarks and outlining our future work.

2 Preliminaries

2.1 SLA Overview

A service level agreement (SLA) [22] is a contract between the service consumer and the service provider, formally defines the level of services. It gives details about the quality and scope of the service provided, which can also be referred to as a "service level contract".

> "A service level agreement, SLA, is a technical contract between two types of businesses, producers and consumers. A SLA captures the agreed upon terms between organizations with respect to quality of service (QoS) and other related concerns. In simple cases, one consumer forms a SLA with a producer." [4]

In service-oriented computing, a SLA is a collection of service level requirements which have been negotiated and mutually agreed upon by the information providers and the information consumers [5]. Usually, providers define some service levels as a fixed combination of their specific capabilities on a set of quality dimensions, and users must choose one these levels. An SLA could be split into different levels, each solving problems for different groups of customers who have the same services, in the same SLA [16], hence the term multilevel SLA as follows.

- Corporate-level SLA: covers all the generic service level management (often abbreviated as SLM) issues appropriate to every customer throughout the organization. These issues are likely to be less volatile and so updates on this kind of SLA are less frequently required.
- Customer-level SLA: addresses all service level issues relevant to a particular customer group, regardless of the services being used.
- Service-level SLA: is relevant to a specific service, in relation to a specific customer group.

Quality of service (QoS) is defined as the "collective effect of service performance, which determines the degree of satisfaction of a user of the service" [15]. However, compatibility with the system and the definition can be used in contracts between service providers and customers, here we define QoS as follows: "QoS is the degree to which service providers can offer customers under contracts have been committed".

One of the most significant QoS concerns of Web services is reliability. For services (cloud) the reliability can be measured by the time the system is ready to serve as intended. For the mission-critical task, the reliability is more important than other aspects [18].

2.2 Semiring

Semiring is a mathematical structure that features a domain equipped with two operations satisfying certain properties, as described in Definition 1.

Definition 1. *A* **semiring** *is a tuple* $\langle A, \oplus, \otimes, \bar{0}, \bar{1}\rangle$ *such that*

- *A is a set and* $\bar{0}, \bar{1} \in A$
- $\oplus$*, called the additive operation, is a commutative, associative operation having* $\bar{0}$ *as its neutral element (i.e.* $a \oplus \bar{0} = a = \bar{0} \oplus a$*)*
- $\otimes$*, called the multiplicative operation, is an associative operation such that* 1 *is its unit element and* $\bar{0}$ *is its absorbing element (i.e.* $a \otimes \bar{0} = \bar{0} = \bar{0} \otimes a$*)*
- $\otimes$ *distributes over* $\oplus$ *(i.e.* $\forall a, b, c \in A \rightarrow a \otimes (b \oplus c) = a \otimes b \oplus a \otimes c$*)*

An idempotent semiring is a semiring whose additive operation is idempotent (i.e. $a \oplus a = a$). This idempotence property allows us to endow a semiring with a canonical order defined as $a \preceq b$ iff $a \oplus b = b$ [7]. There exists another form of idempotent semiring called c-semiring whereby the $\oplus$ operator is defined over subsets of a domain and as such it has flattening property [3]. The endowed order of a c-semiring is actually a partial order that would be used for choosing "best" solutions in a constraint satisfaction problem.

Definition 2 ([2]**).** *A semiring will be called* **c-semiring***, where "c" stands for "constraint", meaning that they are the natural structures to be used when handling constraints. A* **c-semiring** *is a tuple* $\langle A, \oplus, \otimes, \bar{0}, \bar{1}\rangle$ *such that*

- *A is a set and* $\bar{0}, \bar{1} \in A$
- $\oplus$ *is defined over (possibly infinite) sets of elements of A as follows:*
  - *For all* $a \in A, \sum(\{a\}) = a$*;*
  - $\sum(\emptyset) = \bar{0}$ *and* $\sum(A) = \bar{1}$*;*
  - $\sum(\bigcup A_i, i \in I) = \sum(\{\sum(A_i), i \in I\})$ *for all sets of indices I (flattening property).*
- $\otimes$ *is a binary, associative and commutative operation such that 1 is its unit element and 0 is its absorbing element*
- $\otimes$ *distributes over* $\oplus$ *(i.e. for any* $a \in A$ *and* $B \subseteq A$*,* $a \otimes \sum(B) = \sum(\{a \otimes b, b \in B\})$*)*

To make an idempotent semiring applicable for the representation of QoS, we endow it with a canonical order defined as $a \preceq b$ iff $a \oplus b = b$. A semiring is used to express the domain and the order between values that feature a QoS. To represent QoS factors, we may use the notion of *bounded lattice*. Each bounded lattice has a greatest element (denoted as $\top$) and a least element (denoted as $\bot$) and features two operations: meet (denoted as $\wedge$) and join (denoted as $\vee$) [7].

2.3 Business Contract Modeling Based on Deontic Logic

Business contracts specify obligations, permissions and prohibitions as mutual agreements between business parties [19], as well as actions to be taken when a contract is violated. Governatori and Milosevic [12] have proposed such a contract modeling language which includes a non-boolean connective, $\odot$, to represent contrary-to-duty obligations (i.e., what should be done if the terms of a contract are violated). Deontic operators capture the contractual modality (i.e. obligations, permissions and prohibitions) [10]. Governatori et al. represent a contractual rule as $r : A_1, A_2 \dots A_n \vdash C$ where each A_i is an antecedent of the rule and C is the consequent. Each A_i and C may contain deontic operators but connectives can only appear in C.

As an example, $r : \neg p, q \vdash O_{seller}\alpha \odot O_{seller}\beta$ is a contractual rule (identified by r) stating that if antecedents $\neg p$ and q hold, then a seller is obliged to make sure that α is brought about. Failure to do so results in a violation, for which a reparation can be made by bringing about β (the connective $\odot$ can therefore be informally read as "failing which").

Definition 3 (Contractual rules of SLA [12]). *Contractual rules r and r' can be merged into rule r'' as follows where X denotes either an obligation or a permission.*

$$\frac{r : \Gamma \vdash O_s A \odot (\bigodot_{i=1}^{n} O_s B_i) \odot O_s C \qquad r' : \quad \varDelta, \neg B_1, \neg B_2, .., \neg B_n \quad \vdash X_s D}{r'' : \Gamma, \varDelta \vdash O_s A \odot (\bigodot_{i=1}^{n} O_s B_i) \odot X_s D} \tag{1}$$

The $\otimes$ operator is associative but not commutative. This property matters when reasoning about the subsumption and merging of contractual rules. Definition 3 defines how contract rules might be merged. Governatori et al. also devise a machinery for determining if one contractual rule subsumes another as presented in Definition 4.

Definition 4 *(*[12]**).** *Let's consider two rules $r_1 : \Gamma \vdash A \odot B \odot C$ and $r_2 : \varDelta \vdash D$ where $A = \bigodot_{i=1}^{m} A_i$, $B = \bigodot_{i=1}^{n} B_i$ and $C = \bigodot_{i=1}^{p} C_i$. Then r_1 subsumes r_2 (i.e. r_2 can safely be discarded if we have r_1) iff*

1. *$\Gamma = \varDelta$ and $D = A$; or*
2. *$\Gamma \cup \{\neg A_1, \dots, \neg A_m\} = \varDelta$ and $D = B$; or*
3. *$\Gamma \cup \{\neg B_1, \dots, \neg B_n\} = \varDelta$ and $D = A \odot \bigodot_{i=0}^{k \le p} C_i$*

3 Research Motivation

In this section, we first describe a running example[2] (Subsect. 3.1) and then come up with our research statements (Subsect. 3.2).

[2] The running example will be used for explaining our research questions and exemplifying our formality.

3.1 Running Example

Let's consider the business of a tour company that is performed on the theme of Future Internet [20] – a recently-emerging trend that aims to offer integrated access to people, media, services, etc. using an underlying platform. It seeks to enable new styles of social and economic interactions on an unprecedented scale, offering both flexibility and quality. Besides being the constituting building block of the so-called Internet of Services, the Future Internet, through the metaphor of the Internet of Things, will provide location-independent, interoperable, scalable, secure and efficient access to a coordinated set of services. However, to turn the promise of this principle into realized benefits, services must be accompanied by exact definitions as to the conditions of their usage. These conditions can be specified by Service Level Agreement (SLA).

Businesses using SERVQUAL to measure and manage service quality deploy a questionnaire that measures both the customer expectations of service quality in terms of these five dimensions (i.e., reliability, assurance, tangibles, empathy and responsiveness) [9], and their perceptions of the service they receive. When customer expectations are greater than their perceptions of received delivery, service quality is deemed low. According to SERVQUAL, the quality of a business service includes assurance, reliability, responsiveness, tangibles and empathy. Table 1 describes them in detail. Each dimension descriptor has an informal definition in the tourist industry. In our framework, these items together make up the first element to arrange and evaluate the SLA, i.e., satisfaction have a reliability of service and weight of evaluating the importance of a service in the SLA. The other two SLA elements are cost and penalty rules. Figure 1 illustrate our standpoint. When customers

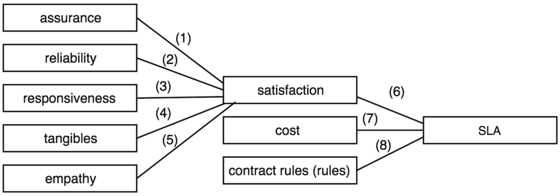


Fig. 1. Breakdown of the SERVQUAL-based SLA

Table 1. Components of SLA in SERVQUAL [9]

Num	Dimensions	Definition
(1)	Assurance	Knowledge and courtesy of employees and their ability to convey trust and confidence
(2)	Reliability	Ability to perform the promised service dependably and accurately
(3)	Responsiveness	Willingness to help customers and provide prompt service
(4)	Tangibles	Physical facilities, equipment, and appearance of personal
(5)	Empathy	Caring, individualized attention the firm provides its customers
(6)	Satisfactions	Satisfaction have a reliability of service and weight of evaluating the importance of a service in the SLA
(7)	Cost	Cost is a payment or price of service
(8)	Rules	The penalty clauses in contract are also represented as contractual rules

book a trip through a travel tour company's website, they get travel information such as destinations, restaurants, hotels. For example:

> "This trip departs within 60 days and requires full payment to confirm a place. Upon receipt of payment we will confirm your booking within 2–4 days. We recommend waiting for our confirmation before purchasing air tickets or other non-refundable travel arrangements."

In Fig. 2, services provided by car rental companies, airlines and hotels will be bundled and offered to customers via a tour company. Customers will interact directly with the system via a booking agency website. This system will be connected to many different types each of which could be provided by at different SLAs by multiple companies. The SLAs will be combined and arranged to send to customers. Customers will choose the SLA have been combined to send to the travel agency. This agency will manage the customer contracts with companies providing services.

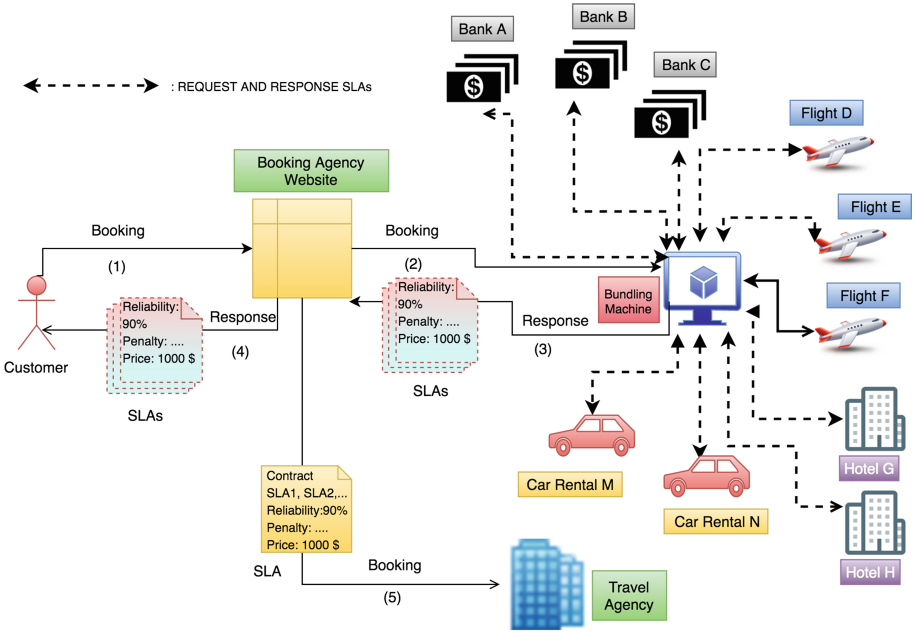


Fig. 2. Bundling car rental agents, airlines and hotels

The tour was made up of a combination SLA of companies providing services as: Bank A, Bank B, Flight D, Flight E, Flight F, Hotel G, Hotel H, Car Rental M, Car Rental N,... Tables 2 and 3 are used to illustrate the service level of the Hotel G, Car rental M in Fig. 2. In fact, each company will release many different SLA and need a screening inspection and combined. Table 2, the Hotel G provided more level services (level SLA) a customer might want as: Luxury (SLA_{H1}), Bussiness (SLA_{H2}), Personal (SLA_{H3}), helping customers to select the one that's best for them. Each level SLA consist of several sub-agreements, the Luxury (SLA_{H1}) level Hotel G include 3 sub-agreement: sa_{h1}, sa_{h2}, sa_{h3}. The sub-agreements is a service, has the weight of evaluating the importance of a services in the SLA. The total weight of a level SLA must equal 1 (100 %), the weight of a sub-agreement must be one minus the weight of another. The total weight of a level SLA = $sa_{h1} + sa_{h2} + sa_{h3} = 1$. Satisfaction of an agreement for a sub-agreements is an operation multiplication between the reliability and weight. In SLA_{H1} level, the satisfaction (sat) of sa_{h1} = weight of sa_{h1} * reliability (rel) of sa_{h1} = 0.2 * 0.99 = 0.198. The total satisfaction of several sub-agreements in the level SLA is satisfaction of the level SLA. The satisfaction of $SLA_{H1} = sa_{h1} + sa_{h2} + sa_{h3}$ =0.198 + 049 + 0.285 = 0.973. The Car rental M provided more level SLA for customer as: Full size(SLA_{R1}), SUV(SLA_{R2}), Economy(SLA_{R3}), the Full size(SLA_{R1}) include 3 sub-agreement: sa_{r1}, sa_{r2}, sa_{r3}. A SLA of a tour is a combination of multiple SLA of 2 Tables 2 and 3 as $\{SLA_{H1}, SLA_{R1}\}, \{SLA_{H1}, SLA_{R1}, SLA_{H1}\}, \{SLA_{H1}, SLA_{R1}, SLA_{R2}\}$, $\{SLA_{H1}, SLA_{R2}\}$:

Table 2. Multi-level SLA for hotel or restaurant

With: Weight(w), Commitment(Com), Reliability (Rel), Satisfactions (Sat)										
Sub-Agreement	w	Luxury(SLA_{H1})			Bussiness(SLA_{H2})			Personal(SLA_{H3})		
		Com	Rel	Sat	Com	Rel	Sat	Com	Rel	Sat
sa_{h1}-Phone	0.2	24/7	0.99	0.198	24/7	0.99	0.198	24/7	0.99	0.198
sa_{h2}-Free Breakfast	0.5	Type1	0.98	0.490	Type2	0.90	0.450	No	0	0
sa_{h3}-Pick up: At airport	0.3	Yes	0.95	0.285	Yes	0.93	0.279	No	0	0
Price (Cost)		$1500		0.973	$1000		0.927	$500		0.198

Table 3. Multi-level SLA for rental car

With: Weight(w), Commitment(Com), Reliability (Rel), Satisfactions (Sat)										
Sub-Agreement	w	Full size(SLA_{R1})			SUV(SLA_{R2})			Economy(SLA_{R3})		
		Com	Rel	Sat	Com	Rel	Sat	Com	Rel	Sat
sa_{r1}-Support of a rental car	0.4	Yes	0.99	0.396	Yes	0.98	0.392	Yes	0.87	0.348
sa_{r2}-Time of delivery car	0.3	Yes	0.98	r0.294	lYes	r0.93	0.279	Yes	0.92	0.278
sa_{r3}-Pick up: At airport	0.3	Yes	0.95	0.285	Yes	0.95	0.285	No	0	0
Price (Cost)		$350		0.975	$300		0.956	$250		0.624

3.2 Research Statements

Let's formulate our research statements based on the running example described above. First, business services usually come with multi-level SLAs. We need a formal representation that captures SLA items described in Table 1. We aim to shorten them to have a more succinct representation for the sake of formal reasoning. In our running example, the representational items in Table 1 will be combined in a way to describe the SLA using no more than 3 elements.

Second, bundling business services involves multiple service providers. As such, we need to represent the multi-level SLAs from at least two perspectives, namely the customers's point of view and the viewpoint of the providers who provide the services to be bundled. In our running example, we take into account the travellers' point of view and the viewpoint of car rental companies, airlines and hotels when in comes to services bundling and SLAs.

Third, bundling business services will lead to the aggregation of their multi-level SLAs. As same service provider may offer multiple business services under different SLAs, bundling services generate a large number of SLAs many of which are multi-level. We need to come up with an approach to systematically handling the generated SLAs and helping customers choose the one that best suits their needs. This is observable in our running example where multiple car rental companies, hotels, etc. make offerings to travellers to pick up a travel package.

4 A Framework for Reasoning About SLA in Services Bundling

4.1 Formal Representation

Our fundamental idea is the definitions to create a SLA document from services of providers. It can be used to create a new SLA from the provider's SLA when they provide services. This means that the relationship of providers and new SLA. Therefore, we can define an new SLA document as follows.

Definition 5. *Each multi-level SLA consist of several sub-agreements. The satisfaction of the SLA is intended to reflect the customer satisfaction, which is the sum of the sub-Agreements's satisfaction* [26]*. Let $S = \sum_{i=1}^{n} s_i$ where S is the* ***Satisfaction*** *of the SLA and s is satisfaction of an agreement of SLA, clearly $S \in [0,1]$.*

The SLA can include many sub-Agreements with different policies defined in each of them. A sub-Agreements (subcontract) is understood as a part of SLA. Each sub-Agreements is a service which has the reliability and weight of evaluating the importance of a services in the SLA. Satisfaction of an agreement for a service is an operation multiplication between the reliability and weight w by the formula: $s = r * w$ Where s is satisfaction of an agreement, r is reliability. The reliability is defined in Difinition 6.

Definition 6. Reliability *is defined as follow: $r = 1 - fr$ where r is reliability of a service, fr is failure rate.*

Reliability of a service, can be represented by the failure rate as shown below and the failure rate of a service can be measured by the ratio of the number of times the service damaged to the total service requests.

Example 1. A customer rented a car which was provided by the company Rental Car. Rental Car contacted the owner determine the time delivery for customers. But all the time the owner had a problem and couldn't deliver for customers. It was recorded as a case of cancelled service.

Definition 7. *Let $C_0 = \{c_1, c_2, ...c_n\}$ be the initial set of values of Cost.* ***Cost*** *is a payment or price of a service which is in the closure of cost values under addition. By closure of C_0, we mean the smallest set containing all summation of elements in C_0: $C_0^+ = \{\sum_{k=1}^{\infty} (C_{i_1} + ... + C_{i_k}) | C_{i_k} \in C_0\}$. Based on those conditions,* ***Cost*** *is defined as $C = C_0^+ \cap [0, Cost_{max}]$, where $Cost_{max}$ is the highest cost that the customer can pay.*

Definition 8. *The complete SLA is a combination of mutilevel SLA from service providers that offer several types of services. The complete SLA is defined* ***SLA (Combining SLAs)*** *which has included the several types of services. Let $K_0 = \{k_1, k_2, ...k_n\}$ be the initial set of types of services in CSLAs. With type of service k_i, We have more service providers to choose SLAs. The service providers will deliver services to make up the initial set $J_0^{k_i} = \{j_1^{k_i}, j_2^{k_i}, ...j_{m_{k_i}}^{k_i}\}$, m_{k_i} is defined l. A set types of services and services of provider is defined as follows: $\{SLA(k_i, j_l^{k_i})\}$ where $k_i \in K_0$ and $j_l^{k_i} \in J_0^{k_i}\}$. Each element of $SLA(k_i, j_l^{k_i})$ is the triple $\langle S, C, R\rangle$,where S: Satisfaction; C: Cost;R: Rule. A* ***CSLA*** *is combined from $SLA(k_i, j_l^{k_i})$ as follows:*

$$\begin{aligned} CSLA &= \bigodot_{i=1}^{n} SLA(k_i, j_l^{k_i}) \\ &= \langle \min S_{SLA(k_i,j_l^{k_i})}, \sum_{k=1}^{n} \left(C_{SLA(k_i,j_l^{k_i})}\right), \mathrm{mergeR}_{\mathrm{SLA(k_i,j_l^{k_i})}}\rangle. \end{aligned} \tag{2}$$

Example 2. The running example in Sect. 3.1, a tour is combinated services' SLA in Tables 2 and 3. Let $K_0 = \{k_1,k_2\}$ where k_1 is the type of service for hotel and k_2 is type of services for the car rental. We call $j_l^{k_1}$ is SLA in the type of service for hotel (k_1). The SLA_{H1} will be converted to $SLA(k_1, j_1^{k_1})$. The conversion of SLAs is presented in Table 4.

Table 4. Combining multi-level SLAs into one CSLA

Services	Provider	Multi-level SLAs are provied for customers			
		Level	SLA	Satisfaction	Cost
Hotel	Hotel G	Luxury(SLA_{H1})	$SLA(k_1, j_1^{k_1})$	0.973	\$1500
		Bussiness(SLA_{H2})	$SLA(k_1, j_2^{k_1})$	0.927	\$1000
		Personal(SLA_{H3})	$SLA(k_1, j_3^{k_1})$	0.198	\$500
Car rental	Car Rental M	Full size(SLA_{R1})	$SLA(k_2, j_1^{k_2})$	0.975	\$350
		SUV(SLA_{R2})	$SLA(k_2, j_2^{k_2})$	0.956	\$300
		Economy(SLA_{R3})	$SLA(k_2, j_3^{k_2})$	0.624	\$250

In Definition 8, the satisfactions of CSLA is the smallest satisfaction in the SLA set. The combination of mutilevel SLA from service providers created a collection of CSLA objects. The table 5 is a range of CSLAs that shows how combinding two or more SLA, which displays the satisfaction index of CSLA.

Table 5. Listing CSLA in details of SLAs

CSLA	Combination SLA	$\langle S, C, R\rangle$
$CSLA_1$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2})$	$\langle 0.975; c; r\rangle$
$CSLA_2$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_2^{k_2})$	$\langle 0.956; c; r\rangle$
$CSLA_3$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; c; r\rangle$
$CSLA_4$	$SLA(k_1, j_2^{k_1}), SLA(k_2, j_1^{k_2})$	$\langle 0.931; c; r\rangle$
$CSLA_5$	$SLA(k_1, j_2^{k_1}), SLA(k_2, j_2^{k_2})$	$\langle 0.931; c; r\rangle$
$CSLA_6$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2}), SLA(k_2, j_2^{k_2}$	$\langle 0.956; c; r\rangle$
$CSLA_7$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; c; r\rangle$
$CSLA_8$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_2^{k_2}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; c; r\rangle$

4.2 Aggregating SLAs

Mathematicaly combining Satisfaction, Cost and Rule results in the set of triple $A = \{\langle S, C, R\rangle\}$, where S is Satisfaction, C is Cost, R is Rule.

- The Satisfaction has become an aspect of quality of service. It has been proven by a number of quality research services related to customer satisfaction: $Satisfaction \in [0, 1]$;
- Cost is a payment or price of service which is presented in Definition 8
- The penalty clauses in contract are also represented as contractual rules. Rules are formulas in first order logic.

We consider the following ordering on set A. Let a $= \langle s_1, c_1, r_1\rangle$ and b $= \langle s_2, c_2, r_2\rangle$. We can say that $a \leq b$ iff $(\mathrm{s}_1 \leq \mathrm{s}_2)$ or $(\mathrm{s}_1 = \mathrm{s}_2) \wedge (\mathrm{c}_1 \leq \mathrm{c}_2)$ or $(\mathrm{s}_1 = \mathrm{s}_2) \wedge (\mathrm{c}_1 = \mathrm{c}_2) \wedge (\mathrm{r}_2 \vdash \mathrm{r}_1)$. In the case that $(s_1 = s_2) \wedge (c_1 = c_2) \wedge (r_2 \nvdash r_1) \wedge (r_1 \nvdash r_2)$, we define $a = b$.

Clearly, the relation "$\leq$" defines a total ordering over the set A. We define the $\oplus$ operation as the max operation with respect to this order.

The $\otimes$ operator is the multiplication acting on each component an element in the set A differently. The $\otimes$ operator's action on S is the min operation. The $\otimes$ operator's action on C is ordinary addition. The $\otimes$ operator's action on R is the merging of two different rules into one rule. More precisely, let $a = \langle s_1, c_1, r_1\rangle$ and $b = \langle s_2, c_2, r_2\rangle$, then $a \otimes b$ is defined as follow

$$a \otimes b := \langle \min\{s_1, s_2\},\ c_1 + c_2,\ \text{merge}(r_1, r_2)\rangle.$$

Proposition 1. *Let* $\bar{0} = \langle \top, 0, 0\rangle$, $\bar{1} = \langle \bot, 0, 1\rangle$, *with* $\top$ *and* $\bot$ *are tautology and the empty set, respectively. Then the tuple* $\langle A, \oplus, \otimes, \bar{0}, \bar{1}\rangle$ *is a semiring.*

Proof. Clearly, $\bar{0}$ is the neutral element of $\oplus$ and $\bar{1}$ is the unit element of $\otimes$. It suffices to show that the $\otimes$ operator is distributive over $\oplus$. Let $a = \langle s_1, c_1, r_1 \rangle$, $b = \langle s_2, c_2, r_2 \rangle$, and $c = \langle s_3, c_3, r_3 \rangle \in A$. We show that $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$.

Indeed, without lost of generality, we may assume that $b \oplus c = c$ (i.e., $b \leq c$) and $b \neq c$. Then we have $r_3 \vdash r_2$. Therefore, $\mathrm{merge}(r_1, r_3) \vdash \mathrm{merge}(r_1, r_2)$. Hence

$$\begin{aligned}
\text{R.H.S.} &= \max(\langle \mathrm{merge}(r_1, r_2),\ c_1 + c_2,\ \min\{s_1, s_2\} \rangle, \\
&\quad \langle \mathrm{merge}(r_1, r_3),\ c_1 + c_3,\ \min\{s_1, s_3\} \rangle) \\
&= \langle \mathrm{merge}(r_1, r_3),\ c_1 + c_3,\ \min\{s_1, s_3\} \rangle \\
&= a \otimes c \\
&= \text{L.H.S.}
\end{aligned}$$

4.3 Sorting

In Example 2, more conditions can be added: the ability to pay of customer, types of services needed for the trip. The ability to pay of customer is $Cost_{max}$ and it can hit $2000. A customer can pick up a tour that comes with an accommodation service and one or two car rental services.

We define a total order for sorting the SLAs in services bundling. This order can intuitively be explained as follows. First, the system looks at CSLAs' satisfaction as a criteria to establish the order. If the system encounters two CSLAs having same satisfaction, the system will look into their cost. The one having a lower cost is preferred to the other. In case the two CSLAs being compared have exactly the same satisfaction and cost, the system will check their penalty rules. Note that penalty rules are sorted by applying the entailment. Formally, we can define this total order as: $CSLA_i \geq CSLA_j\ iff$ $(S_{CSLA_i} > S_{CSLA_j})$ or $(\mathrm{S_{CSLA_i}} = \mathrm{S_{CSLA_j}}) \wedge (\mathrm{C_{CSLA_i}} \leq \mathrm{C_{CSLA_j}})$ or $(\mathrm{S_{CSLA_i}} = \mathrm{S_{CSLA_j}}) \wedge (\mathrm{C_{CSLA_i}} = \mathrm{C_{CSLA_j}}) \wedge (\mathrm{R_{CSLA_i}} \vdash \mathrm{R_{CSLA_j}})$ where S: satisfaction, C: cost, R: rule.

Table 6 illustrates how the first step in this sorting procedure changes the order of the CSLAs listed in Table 5. Note that $CSLA_6$ and $CSLA_3$ have been moved and marked with * to their new location.

Table 6. A CSLA List are ordered by satisfaction

CSLA	Combination SLA	$\langle S, C, R \rangle$
$CSLA_1$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2})$	$\langle 0.975; c; r \rangle$
$CSLA_2$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_2^{k_2})$	$\langle 0.956; c; r \rangle$
* $CSLA_6$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2}), SLA(k_2, j_2^{k_2})$	$\langle 0.956; c; r \rangle$
$CSLA_4$	$SLA(k_1, j_2^{k_1}), SLA(k_2, j_1^{k_2})$	$\langle 0.931; c; r \rangle$
$CSLA_5$	$SLA(k_1, j_2^{k_1}), SLA(k_2, j_2^{k_2})$	$\langle 0.931; c; r \rangle$
* $CSLA_3$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; c; r \rangle$
$CSLA_7$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; c; r \rangle$
$CSLA_8$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_2^{k_2}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; c; r \rangle$

After finding the CSLA with equal satisfaction, the system continue to add the cost of CSLA to Table 6. The CSLA's cost are calculated by the formula $C = C_0^+ \cap [0, Cost_{max}]$ in the Difinition 7. By comparing the maximum value of the two closures, we will choose the better CSLA as follows: $C_{CSLA_1} > C_{CSLA_2}$ iff $\max\{\mathrm{C}^+_{\mathrm{CSLA}_1} \cap [0, \mathrm{Cost}_{\max}]\} < \max\{\mathrm{C}^+_{\mathrm{CSLA}_2} \cap [0, \mathrm{Cost}_{\max}]\}$. In Table 6, we have:

$$\begin{aligned} CSLA_5 > CSLA_4 &\Leftrightarrow \max\{C^+_{CSLA_5} \cap [0, Cost_{max}]\} < \max\{C^+_{CSLA_4} \cap [0, Cost_{max}]\} \\ &\Leftrightarrow \max\{C^+_{CSLA_5} \cap [0, 2000]\} < \max\{C^+_{CSLA_4} \cap [0, 2000]\} \\ &\Leftrightarrow 1900 < 2000. \end{aligned}$$

Table 7 illustrates how the second step in our sorting procedure makes a new order between the CSLAs. If CSLAs changed positions they will be marked **.

Table 7. A CSLA List are ordered by cost

CSLA	Combination SLA	$\langle S, C, R \rangle$
$CSLA_1$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2})$	$\langle 0.975; 1500, 350; r \rangle$
$CSLA_2$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_2^{k_2})$	$\langle 0.956; 1500, 300; r \rangle$
$*\ CSLA_6$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2}), SLA(k_2, j_2^{k_2})$	$\langle 0.956; 1500, 350, 300; r \rangle$
$**CSLA_5$	$SLA(k_1, j_2^{k_1}), SLA(k_2, j_2^{k_2})$	$\langle 0.931; 1000, 300; r \rangle$
$CSLA_4$	$SLA(k_1, j_2^{k_1}), SLA(k_2, j_1^{k_2})$	$\langle 0.931; 1000, 350; r \rangle$
$**CSLA_8$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_2^{k_2}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; 1500, 300, 250; r \rangle$
$*\ CSLA_3$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; 1000, 250; r \rangle$
$CSLA_7$	$SLA(k_1, j_1^{k_1}), SLA(k_2, j_1^{k_2}), SLA(k_2, j_3^{k_2})$	$\langle 0.624; 1000, 350, 250; r \rangle$

5 Related Work

In general, SLA management has been studied in the past few years and it has been mainly concentrated on the definition of languages and the specification of standards for SLA [6,23]. However, these standards are still evolving as they present some limitations. The existing frameworks for SLA management only define the format and types of messages that can be exchanged during the negotiation between service providers and service consumers. Some projects particularize the management of SLAs to specific domains, such as military, database management, or information systems [8,13,25]. Garvin proposed a quality management grid featuring a total of eight dimensions including performance and reliability [11].

In our paper, we provide insights of how the SLAs of business services could be aggregated in services bundling. Our approach to combining SLAs is focused on cost, reliability and penalty. Our goal is to provide a mechanism for sorting the SLAs generated.

6 Conclusions

The representation of business services requires that we view human activity and human-mediated functionality through the lens of computing and systems engineering. Services bundling is a practice that creates new service values by purposely combining business services. The resutling combination, called a new service offering, poses a few challenging questions of how to aggregate the SLAs of the bundled services. We give insights into the modeling of SLA for high-level business services taking into account their human-mediating nature. Concretely, the SLA should be multi-level and incoorporate technical attributes such as reliability and contract-like statements such as payment and penalty rules. We use Deontic logic for formally reasoning about penalty rules. Altogether, we leverage the mathematical structure of semiring to represent the SLA as a whole, which helps explain the intuitive meaning of aggregating the SLAs of business services when bundling them.

Future Investigations. Our future work includes: (i) devising algorithms for generating the SLAs when bundling services; (ii) designing a recommendation mechanism that would suggest the service consumers pick up an aggregated SLA according to their service preference.

References

1. Bishop, K., Bolan, G., Bowen, D., Cromack, C., Evans, S., Fisk, R.P., Ganz, W., Gregory, M., Johnston, R., Lemmink, J., et al.: Succeeding through service innovation: a service perspective for education, research, business and government (2008)
2. Bistarelli, S.: Semirings for soft constraint solving and programming. In: Bistarelli, S. (ed.) Semirings for Soft Constraint Solving and Programming. LNCS, vol. 2962, pp. 1–20. Springer, Heidelberg (2004)
3. Bistarelli, S., Montanari, U., Rossi, F.: Semiring-based constraint satisfaction and optimization. J. ACM (JACM) **44**(2), 201–236 (1997)
4. Blake, M.B., Cummings, D.J., Bansal, A., Bansal, S.K.: Workflow composition of service level agreements for web services. Decis. Support Syst. **53**(1), 234–244 (2012)
5. Cappiello, C., Comuzzi, M.: Efficient allocation of quality improvement efforts to support the definition of data service offerings. In: Proceedings of the 12th International Conference on Information Quality (ICIQ 2007), pp. 209–220 (2007)
6. Czajkowski, K., Foster, I., Kesselman, C., Sander, V., Tuecke, S.: SNAP: a protocol for negotiating service level agreements and coordinating resource management in distributed systems. In: Feitelson, D.G., Rudolph, L., Schwiegelshohn, U. (eds.) JSSPP 2002. LNCS, vol. 2537, pp. 153–183. Springer, Heidelberg (2002)
7. Davey, B.A., Priestley, H.A.: Introduction to Lattices and Order. Cambridge University Press, Cambridge (2002)
8. Falkowski, T., Voß, S.: Application service providing as part of intelligent decision support for supply chain management. In: Proceedings of the 36th Annual Hawaii International Conference on System Sciences. IEEE Computer Society (2003)
9. Fick, G.R., Ritchie, J.B.: Measuring service quality in the travel and tourism industry. J. Travel Res. **30**(2), 2–9 (1991)

10. Gabbay, D.M., Woods, J.: Logic and the Modalities in the Twentieth Century. Handbook of the History of Logic, vol. 7 (2006)
11. Garvin, D.A.: Managing Quality: The Strategic and Competitive Edge. Simon & Schuster, New York (1988)
12. Governatori, G., Milosevic, Z.: A formal analysis of a business contract language. Int. J. Coop. Inf. Syst. **15**(04), 659–685 (2006)
13. Greenwood, D., Vitaglione, G., Keller, L., Calisti, M.: Service level agreement management with adaptive coordination. In: Proceedings of the International Conference on Networking and Services (ICNS 2006), pp. 45–45. IEEE Computer Society (2006)
14. Hansmann, U., Merk, L., Nicklous, M.S., Stober, T.: Pervasive Computing: The Mobile World. Springer, Heidelberg (2003)
15. ITU: ITU-T E.800 E.800: Definitions of Terms Related to Quality of Service (2008)
16. Jineja, R., Sharma, D.: Multi-level SLA management architecture for cloud computing. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **4**(5), 401–406 (2014)
17. Lê, L.-S., Ghose, A.: Contracts + goals = roles? In: Atzeni, P., Cheung, D., Ram, S. (eds.) ER 2012 Main Conference 2012. LNCS, vol. 7532, pp. 252–266. Springer, Heidelberg (2012)
18. Li, J., Tang, W., Wang, X.: Adding QoS to web service transaction management. In: Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems, vol. 2, pp. 1308–1313. IEEE Computer Society (2004)
19. Linington, P.F., Milosevic, Z., Cole, J., Gibson, S., Kulkarni, S., Neal, S.: A unified behavioural model and a contract language for extended enterprise. Data Knowl. Eng. **51**(1), 5–29 (2004)
20. Pan, J., Paul, S., Jain, R.: A survey of the research on future internet architectures. IEEE Commun. Mag. **49**(7), 26–36 (2011)
21. Paulson, L.D.: Services science: a new field for today's economy. Computer **39**(8), 18–21 (2006)
22. Philip, P.B., Lewis, G., Merson, P.: Service level agreements in service-oriented architecture environments. Technical Note of Software Engineering Institute (2008)
23. Rahwan, I., Kowalczyk, R., Pham, H.H.: Intelligent agents for automated one-to-many e-commerce negotiation. In: Proceedings of the 25th Australasian Conference on Computer Science, pp. 197–204. Australian Computer Society, Inc. (2002)
24. Singh, M.P., Huhns, M.N.: Service-Oriented Computing: Semantics, Processes, Agents. Wiley, Hoboken (2006)
25. Yan, J., Kowalczyk, R., Lin, J., Chhetri, M.B., Goh, S.K., Zhang, J.: Autonomous service level agreement negotiation for service composition provision. Future Gener. Comput. Syst. **23**(6), 748–759 (2007)
26. Yang, S.J., Zhang, J., Lan, B.C.: Service level agreement-based QoS analysis for web services discovery and composition. Int. J. Internet Enterp. Manag. **5**(1), 39–58 (2006)

Non-disjoint Multi-agent Scheduling Problem on Identical Parallel Processors

F. Sadi[1,2], T. Van Ut[1,3], N. Huynh Tuong[4], and A. Soukhal[1](✉)

[1] Laboratory of Computer Science (EA 6300), Team Recherche Opérationnelle,
Ordonnacement et Transport ROOT ERL-CNRS 6305), 64 Avenue Jean Portalis,
37200 Université François Rabelais Tours, France
sadi.faiza@gmail.com, ameur.soukhal@univ-tours.fr

[2] INSA Centre Val de Loire, 3 rue de la chocolaterie, 41000 Blois, France

[3] Can Tho University of Technology (CTUT), 256 Nguyen Van Cu Street,
Ninh Kieu District, Can Tho City, Vietnam
vanut.tran@etu.univ-tours.fr

[4] Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology, VNU-HCM,
268 Ly Thuong Kiet Street, Ho Chi Minh City 740500, Vietnam
htnguyen@hcmut.edu.vn

Abstract. Scheduling problems in which agents (users, customers, application masters, resource manager, etc.) have to share the same set(s) of resources are at the frontier of combinatorial optimization and cooperative game theory. This paper deals with scheduling problems arising when two agents, each with a set of nonpreemptive jobs, compete to perform their respective jobs on two common identical parallel machines. Each agent aims at minimizing a certain objective function that depends on the completion times of its jobs only. The objective functions we consider in our study are makespan and number of tardy jobs. The agents may share some jobs and this problem is called *non-disjoint multi-agent scheduling problem* [3]. Finding the optimal solution for one agent with a constraint on the other agent's cost function is known to be $\mathcal{NP}$-hard. To obtain best compromise solutions for each agent, we propose polynomial and pseudo-polynomial heuristics. Two mixed integer linear programming models are developed to calculate exact non-dominated solutions. Experimental results are conducted to measure the solutions quality given by heuristics.

Keywords: Multicriteria optimization · Multiagent scheduling · Parallel processors · Heuristics · Dynamic programming · Linear mathematical programming

1 Introduction

Efficient management of large-scale job processing systems is a challenging problem, particularly in the presence of multi-users. In addition, real world systems require processing of jobs on common resources and considering different

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 400–414, 2016.
DOI: 10.1007/978-3-319-48057-2_28

objective functions. Most conventional algorithms are designed for multi-criteria scheduling problems where each measure is applied on the whole set of jobs without any distinction. However, these classical multi-criteria scheduling problems are not well appropriated to the more general resource allocation problem across heterogeneous networks frequently encountered in real applications. It is useful to classify jobs as either time constrained, which should be scheduled as soon as possible, or non-time-constrained, which should simply be processed before their due date. This is the subject of multi-agent scheduling [3,8,12]. Indeed, in [11] the author considered an integrated-services packet-switched networks such as ATM (Asynchronous Transfer Mode). Information carried by the network are first split into smaller messages called *packets*. The data comes from different types, such as voice, video, image and so on. Each packet is wrapped with the essential information needed to get it from its source to the correct destination. In the case of audio and video data the authors show that minimizing the number of late delivered packets is more relevant when for the other types of data, the minimizing delay queuing is more suitable. Delay queuing is commonly expressed in the scheduling literature by the total completion time. Peha's results provide polynomial time algorithms to schedule a set of n jobs on m identical parallel machines with assumption of unit processing times.

In this paper, we study the problem of scheduling jobs on identical parallel processors. The model is featured by agents (local decision makers) - each of which is associated with a subset of jobs to perform, and each one has its own objective function depending only on the completion times of its jobs. The agents share not only the common resources but also some common jobs. This problem has been introduced by [3] and called *non-disjoint multi-agent scheduling problem*. These problems belong to a particular class of multi-criteria scheduling problems where their practical and theoretical benefits are highlighted in [3]. During this past decade, such a class has drawn a significant interest to researchers dealing with scheduling problems and from operational research domain.

Depending on the agent's relationships, three scenarios have been defined in [3]: COMPETING (CO), INTERFERING (IN) and NON-DISJOINT (ND). According to our knowledge, except few results appeared in [3], the non-disjoint scenario is not already studied in the literature. However, the competing scenario is no doubt the most studied scenario until now. It was introduced in [4], the authors considered two disjoint sets of jobs, each one is associated with one agent and one objective function. The jobs have to be executed on the single machine and the goal is to find the best compromise solutions between the two agents. When the ε-constraint approach is used, a polynomial time algorithm is proposed to minimizing the number of tardy jobs of each agent. In [10], the authors study Peha's problem introduced in [11] by considering any processing times (not necessary identical). They present an NP-hardness proof and propose a dynamic programming algorithm to calculate a non-dominated solution. The interfering job set scheduling problems is particularly studied in [8,12]. Polynomial and pseudo-polynomial time algorithms are derived for settings with various combinations of the objective functions in the case of single processor and parallel processors.

The rest of this paper is organized as follow. We define the problem in Sect. 2. In Sect. 3, we propose a discussion on the appropriate solutions structure. Section 4 is dedicated to the mathematical programming formulations which determine strict pareto solutions. Two polynomial heuristics are proposed and presented in Sect. 5. We also develop two pseudo-polynomial heuristics in Sect. 6. A comparison between the exact and approximate methods is illustrated in Sect. 7. Conclusion and future researches are presented in Section 8.

2 Problem Definition and Notations

We consider two competitive agents A and B sharing the same machines. Each agent is owning a set of jobs. We denote by $\mathcal{J}^A = \{J_1^A, J_2^A, \ldots, J_{n_A}^A\}$ a job set associated with agent A, while $\mathcal{J}^B = \{J_1^B, J_2^B, \ldots, J_{n_B}^B\}$ is the job set associated with agent B. The agents can share some jobs, that means that $\mathcal{J}^A \cap \mathcal{J}^B$ is not necessary empty. The whole set of jobs is denoted by $\mathcal{J}$ such as $|\mathcal{J}| = n$, that is given by $\mathcal{J}^{\mathcal{A}} \cup \mathcal{J}^B$. The machine environment is composed of two identical parallel machines, denoted by M_i that are always available, for $i = 1, 2$. Preemption is not allowed and each machine can process only one job at a time. The processing time of J_j denoted by p_j is known and given, where C_j is its completion time, $\forall j \in \mathcal{J}$. We assume that all jobs are available at time zero, and jobs within agent B are subject to a common due date denoted by d^B. In this study, the objective function to be minimized of agent A is the makespan of its jobs: $C_{\max}^A = \max_{J_j^A \in \mathcal{J}^A} C_j$, while agent B minimizes the number of its tardy jobs: $\sum U_j^B = \sum_{J_j^B \in \mathcal{J}^B} 1_{\left\{C_j > d^B\right\}}$.

According to the three fields notation of the multi-agent scheduling problems introduced by [3], the problem addressed in this paper is denoted by $P2|ND, d^B|(C_{max}^A, U_j^B)$, where ND means that the job sets are non-disjoint. When the ε-constraint approach is considered, the studied problem is denoted by $P2|ND, d^B, \sum U_j^B \leq Q_B|C_{max}^A$. In this case the problem is to find a schedule that minimizes the objective function of agent A, while keeping the objective function of agent B less than a given threshold Q_B.

3 Structure of the Non-dominated Solutions

Schedule σ is called Pareto solution if there does not exist another solution that dominates it. On the basis of the studied problem properties, we want to determine the overall structure of the Pareto solutions. Some of these properties are generalization of classical single machine scheduling problems. In fact, as jobs of agent B are submitted to one common due date, it is easy to see that they have to be sequenced on each machine according to their shortest processing time order, thus to minimize the number of tardy jobs of agent B. It can also be shown that on a given machine, the tardy jobs of agent B that belong to

$\mathcal{J}^B \backslash \{\mathcal{J}^A \cap \mathcal{J}^B\}$, have to be scheduled last, otherwise the makespan of the agent A can be increased. So, we can write the following proposition.

Proposition 1 *If problem $Pm|ND, d^B, \sum U_j^B \leq Q_B|C_{max}^A$ admits a feasible solution, then it is possible to build an optimal solution such that on each machine we have:*

1. *Jobs of agent B appear in SPT order.*
2. *Tardy job J_j within $\mathcal{J}^B \backslash \{\mathcal{J}^A \cap \mathcal{J}^B\}$ is scheduled after the jobs of agent A.*

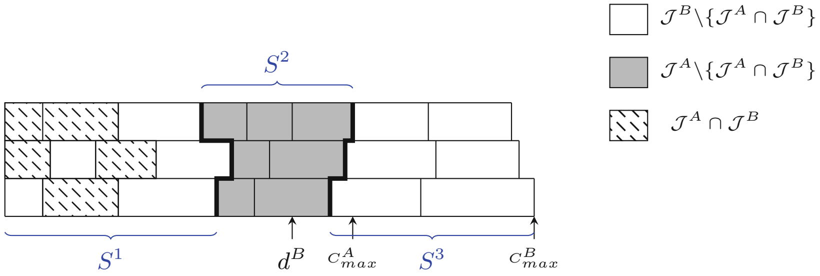

S^2
S^1
d^B
C_{max}^A
S^3
C_{max}^B
$\mathcal{J}^B \backslash \{\mathcal{J}^A \cap \mathcal{J}^B\}$
$\mathcal{J}^A \backslash \{\mathcal{J}^A \cap \mathcal{J}^B\}$
$\mathcal{J}^A \cap \mathcal{J}^B$

Fig. 1. Structure of non-dominated solution for the $Pm|ND, d^B|C_{max}^A, \sum U_j^B$.

Notice that the jobs of agent B should be scheduled before the common due date D^B. Thereby, to minimize the number of tardy jobs (maximize the number of early jobs) we only have to schedule jobs according to SPT rule. Therefore, there is an optimal schedule, if there exists, with structure similar to the one presented in Fig. 1: a sub-schedule S^1 containing jobs of agent B and sequenced according to SPT order on each machine, followed by a sub-schedule S^2 consisting of the remaining jobs of agent A, ended by a sub-schedule S^3 including the remaining jobs within agent B (some agent B jobs in S^3 may be early jobs).

4 Integer Programming Formulation

We present in this section two mixed integer linear programming formulations to solve the ϵ-constraint non-disjoint multi-agent scheduling problem. These models are tailored for the problem considering m identical parallel machines ($m > 0$). The first type of our proposed mathematical model is based on assignment of the jobs to machines, it is presented in Sect. 4.1. The second formulation is based on the time indexed, it is presented in Sect. 4.2.

To generate the set of strict Pareto solutions (Pareto front), we solve the problem with different values of Q_B. At the first iteration, Q_B is set to the upper bound of the objective function $\sum U_j^B$ (at worst case this value is equal to n_B). The obtained solution is denoted by $(\hat{\alpha}^A, \hat{\alpha}^B)$. We then solve the symmetric problem, denoted by $Pm|C^A_{max} \leq \hat{\alpha}^A|\sum U^B_j$. The obtained solution is hence strict Pareto solution, denoted by $(\hat{\alpha}^A, \hat{\alpha}^{B'})$. It is then added to the current set of non-dominated solutions. Next we iterate with $Q_B = \hat{\alpha}^{B'} - 1$; If no feasible solution is obtained then the procedure is stopped.

4.1 Assignment-Based Formulation

Let us consider the following decision variables:

- $x_{i,j}$ is a binary variable that takes value 1 if job J_j is scheduled on machine M_i; 0 otherwise.
- $y_{j,k}$ is a binary variable that is equal to 1 if job J_j is executed before job J_k on the same machine; 0 otherwise.
- z_j is a binary variable that is equal to 1 if job J_j is scheduled after its due date d_B; 0 otherwise.

We also need to define continuous variables: C_j is the completion time of job J_j and C^A_{max} is the value of the agent A objective function. We define HV some positive high value.

$$(MILP - Assign) \quad Min \quad C^A_{max}$$

$$s.t. \begin{cases} \sum_{i=0}^m x_{i,j} & = 1, \quad \forall J_j \in \mathcal{J} & (1) \\ C_j - \sum_{k=1}^n p_k \times y_{j,k} & \geq p_j, \quad \forall J_j \in \mathcal{J} & (2) \\ y_{j,k} + y_{k,j} & \leq 1, \quad \forall J_j, J_k \in \mathcal{J} & (3) \\ x_{i,j} + x_{i,k} - y_{j,k} - y_{k,j} & \leq 1, \quad \forall J_j, J_k \in \mathcal{J} & (4) \\ & \qquad i = 1, \ldots, m & \\ x_{i,j} + y_{j,k} - x_{i,k} & \leq 1, \quad \forall J_j \in \mathcal{J} & (5) \\ & \qquad i = 1, \ldots, m & \\ y_{j,k} + y_{k,l} - y_{j,k} & \leq 1, \quad \forall J_j, J_k, J_l \in \mathcal{J} & (6) \\ C_j - d_B - HV z_j & \leq 0, \quad \forall J_j \in \mathcal{J}^B & (7) \\ \sum_{J_j \in \mathcal{J}^B} z_j & \leq Q_B & (8) \\ x_{i,j}, y_{j,k}, z_j \in \{0,1\}, C_j \geq p_j & \qquad \forall J_j, J_k \in \mathcal{J} & (9) \\ & \qquad i = 1, \ldots, m & \end{cases}$$

Constraints (1) impose that each job J_j should be assigned to one and only one machine. Constraints (2) mean that each jobs' completion time is at least equal to the summation of the processing times of the jobs scheduled before it on the same machine plus its own processing time. Constraints (3) avert solutions that consider job J_j scheduled before job J_k, concurrently job J_k scheduled before job J_j. Constrains (4) impose that if two jobs are on the same machine, then one must be scheduled before another. Constraints (5) express that if J_j is executed on machine M_i and J_j is before J_k, than J_k must be scheduled on machine M_i also. Transitivity constraints are expressed by formula (6), which mean that if J_j is executed before J_k, and J_k is executed before J_l than J_j is executed before J_l. Constraints (7) fixe the values of the binary variables z_j: if it equals to 0 then we have $C_j \leq d^B$, otherwise the job is late and the constrain is still valid. Constraints (8) express the ε-approach bound and constraints. Constraints (9) are the integrity ones.

4.2 Time-Based Formulation

In this section, we propose a second mathematical formulation to solve the considered scheduling problem. We consider $s_{j,t}$ a new binary variables that are time indexed. $s_{j,t}$ takes as a value 1 if job J_j starts its processing at time t; 0 otherwise. Thereby, we have $n \times (T+1)$ binary variables, which is pseudo-polynomial. The general formulation is the following one.

$$(MILP - Time) \;\; Min \;\; C^A_{max}$$

$$s.t. \begin{cases} \sum_{t=0}^{T} s_{j,t} = 1, & \forall J_j \in \mathcal{J}^A \quad (10) \\ \sum_{J_j \in \mathcal{J}} \sum_{l=\max\{0,t-p_j+1\}}^{t} s_{j,l} \leq m, & \forall t = 0, \ldots, T \quad (11) \\ C^A_{max} - \sum_{t=0}^{T-p_j} (t+p_j) s_{j,t} \geq 0, & \forall J_j \in \mathcal{J}^A \quad (12) \\ \sum_{t=0}^{T-p_j} (t+p_j) s_{j,t} - HV z_j \leq d^B, & \forall J_j \in \mathcal{J}^B \quad (13) \\ \sum_{J_j \in \mathcal{J}^B} z_j \leq Q_B, & (14) \\ s_{j,t} \in \{0,1\}, z_j \in \{0,1\}, & \forall J_j \in \mathcal{J} \quad t = 0, \ldots, T, \; (15) \end{cases}$$

Constraints (10) impose that each job J_j starts at a given time t. Constraints (11) ensure that no more than m jobs are scheduled simultaneously, it avoids the jobs overlapping at any time t. Constraints (12) determine the makespan value of jobs of agent A. Constraints (13) fixe the values of the binary variables z_j: if $z_j = 0$ then we have $C_j \leq d^B$, otherwise job J_j is late and the constrain is still valid. Constraints (14) express the ε-approach bound. Constraints (15) are the integrity ones.

5 Polynomial Heuristics

The studied scheduling problems are NP-hard [12], we propose computationally efficient heuristics technical. To illustrate our approach, let consider the case of two machines ($m = 2$). Our approach can be easily extended to multiple processors environment. Our goal is to generate the set of non-dominated solutions, where the decision maker can evaluate the tradeoffs in the criteria. This is a posteriori approach in which the decision maker makes his choice only after a set of points is presented. Of course, our resolution methods can be also used as interactive approach where the decision maker specifies the maximum allowed number of tardy jobs of agent B and seek for the optimal value for the makespan of agent A.

Since makespan is equivalent to the decision problem involving a common deadline (i.e. whether a feasible schedule can be obtained such that all jobs finish before a common deadline) the set of non-dominated solutions can give the decision-maker important information on whether jobs for one set can be finished by a given time and the resulting compromise or effect on the number of tardy jobs for the other agent.

We consider the problem where the agent B objective function is bounded by Q_B. The goal is to obtain at least $n_B - Q_B$ early jobs. Since Property 1 specifies that the jobs of agent B have to be sequenced in their smaller processing time order, we consider in the following subset of jobs E containing the smallest $n_B - Q_B$ jobs of agent B. These jobs should be scheduled early so as to have a feasible solution. In order to obtain an approximate Pareto front, we solve the problem with different values of Q_B, such that $Q_B \in [0, n_B]$. Then after, dominated solutions are removed.

5.1 Heuristic 1

We focus on minimizing the makespan of agent A, using the heuristic *LPT-FAM* that is a well heuristic method for solving classical scheduling problem $Pm||C_{\max}$. It sorts the jobs in non-increasing processing times order (LPT) and assigns them to the first available machine (FAM).

The heuristic presented in this section is made up of three phases, where *LPT+FAM* is used (see Algorithm 1). It starts by scheduling jobs of E, followed by the remaining jobs of agent A from $\mathcal{J}^A \backslash \{\mathcal{J}^A \cap E\}$, and finishes by sequencing jobs of agent B not already scheduled. The remaining jobs of agent B have no influence on the makespan we can execute them at the end of any machine, but in order to optimize the number of tardy jobs it would be more efficient to use *LPT+FAM*. This heuristic is presented in Algorithm 1 and has $O(n_A log(n_A) + n_B log(n_B))$ time complexity.

5.2 Heuristic 2

In this section we develop the precedent heuristic by allowing the jobs reassignment. Indeed, instead of scheduling the jobs using *LPT-FAM*, this heuristic tries progressively to schedule jobs of $E \cup \mathcal{J}^A$, so as to minimize the makespan, and when a job of E is scheduled late, then it is rescheduled until is executed early. Alternatively, this heuristic cannot find a feasible solution. The complexity is still $O(n_A log(n_A) + n_B log(n_B))$, since the step 8 of Algorithm 2 can be done in constant time.

Algorithm 1. LPT-FAM

```
Sort jobs in $J^B$ according to SPT rule;
Set $E = \{J_1^B \dots , J_{n_B - Q_B}^B\}$;
Schedule the jobs in $E$ using $LPT\text{-}FAM$;
if at least one job is late then
    Stop; // this heuristic cannot find a feasible solution
else
    Schedule jobs in $\mathcal{J}^A \backslash \{\mathcal{J}^A \cap E\}$ using $LPT\text{-}FAM$;
    Schedule jobs in $\mathcal{J}^B \backslash \{\mathcal{J}^B \cap E\}$ using $LPT\text{-}FAM$;
    Return the resulting solution;
```

Algorithm 2. LPT-FAM with jobs rescheduling

```
Sort the jobs in $J^B$ according to SPT rule;
Set $E = \{J_1^B \dots , J_{n_B - Q_B}^B\}$;
Set $S = E \cup \mathcal{J}^A$ in LPT order;
Set $E^B = 0$; // the number of early jobs;
while $S \neq \emptyset$ and $E^B < n_B - Q_B$ do
    Schedule $J_j$ using $LPT\text{-}FAM$;
    if $J_j$ is late then
        Remove largest job $J_k$ already scheduled, $J_k \notin E$;
        Put $S = S \backslash \{J_k\}$
        Reschedule $J_j$ using $LPT\text{-}FAM$;
    else
        Set $E^B = E^B + 1$;
if $E^B = n_B - Q_B$ then
    Schedule jobs of $\mathcal{J}^A$ not already scheduled using $LPT\text{-}FAM$;
    Return the resulting solution;
else
    Stop; // This heuristic cannot find a feasible solution;
```

6 Pseudo-polynomial Heuristics

In general, classical heuristics for solving scheduling problems are greedy algorithms based on priority rules. These approaches guaranty an efficient computational time but local optimality. In this section we try to overcome the gap by solving in the exact way the problem minimizing the makespan, that is $P2||C_{max}(S)$ such as $S \subseteq \{\mathcal{J}^A \cup \mathcal{J}^B\}$. We will use the following pseudo-polynomial time algorithm based on dynamic programming [6]:

Let P_j the makespan on machine M_i (i=1,2), and $F_j(P_1, P_2)$ is a recursive boolean function, it is equal to *true* if jobs $J_1, \dots , J_j$ of S can be scheduled on M_1 and M_2 in such away that each machine M_i is busy in interval $[0, P_i]$ $(i = 1, 2)$, and $false$ otherwise.

Applying $F_0(0, 0) = true$ and $F_0(P_1, P_2) = false\ \forall (P_1, P_2) \neq (0, 0)$. $F(j, t_j) = min(F(j-1, p_j - t_j); F(j-1, t_j) + p_j)$, the recursive function is given as follows:

$$F_j(P_1, P_2) = \big(F_j(P_1 - p_j, P_2) \wedge F_j(P_1, P_2 - p_j)\big),$$

$$\forall j = 1, \ldots, n,\ \forall P_i \in [0, UB], \forall i = 1, 2.$$

This dynamic programming algorithm (DP) determines the assignment of jobs to machines, which is sufficient to compute an optimal schedule that minimizes a makespan. The optimal makespan value is given by

$$C_{max}(S) = \min(\max_{\forall P_i \in [0,UB]}(\{P_1, P_2\}|F_j(P_1, P_2) = true)$$

This algorithm runs in $O(nUB^2)$ time, where UB is the upper bound of the makespan.

6.1 Heuristic 3

This heuristic start by scheduling the $n_B - Q_B$ first jobs of agent B, by applying the previous DP. If the optimal value of the obtained makespan is greater than d_B then there is no feasible solution for the problem. Otherwise jobs are scheduled early and they are followed by the remaining jobs of agent A and after by the remaining jobs of agent B, by always applying this DP. This algorithm runs in $O(n^2 + nUB^2)$ time, where UB is the upper bound of the makespan.

Algorithm 3. Heuristic based on dynamic programming

```
Sort the jobs in $J^B$ according to SPT rule;
Set $E = \{J_1^B \ldots, J_{n_B - Q_B}^B\}$;
Optimally solve problem $P2||C_{max}$ considering only jobs of $E$ by the DP
if $C_{max}(E) > d^B$ then
    Stop; // This problem has no feasible solution;
Optimally solve problem $P2||C_{max}$ considering only jobs of $(J^A\setminus\{J^A \cap E\})$ by the DP and taking into account no-availability machines at time zero (jobs of $E$ have been already scheduled)
Optimally solve problem $P2||C_{max}$ considering only jobs of $(\mathcal{J}^B\setminus\{\mathcal{J}^A \cup E\})$ by the DP and taking into account no-availability machines at time zero (previous jobs have been already scheduled)
Try to schedule tardy jobs of agent $B$ earlier without increasing makespan value by moving them to the left before $d^B$
Return the resulting solution;
```

6.2 Heuristic 4

This heuristic is inspired from Heuristic 2, but instead of using *LPT-FAM* for assigning the jobs, it uses the dynamic program presented above. This algorithm runs in $O(nlogn + nUB^2)$ time, where UB is the upper bound of the makespan.

Algorithm 4. LPT-FAM-Dynamic programming

```
Sort the jobs in $J^B$ in SPT order;
Set $E = \{J_1^B \ldots, J_{n_B - Q_B}^B\}$;
Set $S = E \cup \mathcal{J}^A$ in LPT order;
Set $E^B = 0$; // the number of early jobs;
while $S \neq \emptyset$ and $E^B < n_B - Q_B$ do
    Schedule $J_j$ using LPT-FAM;
    if $J_j$ is late then
        Remove $J_k$ the largest job already scheduled;
        Put $S = S \backslash \{J_k\}$
        Reschedule $J_j$ using LPT-FAM;
    else
        Set $E^B = E^B + 1$;
if $E^B = n_B - Q_B$ then
    if some jobs of $E^B$ are late then
        Stop; // This problem has no feasible solution;
    else
        Use DP to schedule the jobs of $\mathcal{J}^A$ not already scheduled;
        Use DP to schedule the jobs of $\mathcal{J}^B$ not already scheduled;
Return the resulting solution;
```

7 Computational Results

The algorithms under study are coded in C language and executed on a workstation with a 2.4 GHz Intel Core i5 processor with 8 GB of memory running Mac OS X Lion 10.7.5. Cplex version 12.6.2 was used to solve the mathematical model. The computation time limit has been fixed to 3600 s for each value (getting only one Pareto solution), where the Pareto fronts are determined as described in Sect. 4.

In this section, we compare Pareto fronts generated by each heuristic presented in Sects. 5 and 6 to the exact Pareto fronts generated by the mathematical integer linear programs (presented in Sect. 4). The used instances are generated with different settings. Processing times, are generated using a discrete uniform distribution from 1 to 10. For each value of n such that $n \in \{10, 20, 30, 40, 50, 70\}$, thirty instances are generated. For each instance, the jobs are assigned randomly to the agents. We generate uniformly a value in the interval $[1, 3]$, for the value 1, the job belongs to $\mathcal{J}^A \backslash \{\mathcal{J}^A \cap \mathcal{J}^B\}$; 2, the job belongs to $\mathcal{J}^A \backslash \{\mathcal{J}^A \cap \mathcal{J}^B\}$ and 3, the job belongs to $\{\mathcal{J}^A \cap \mathcal{J}^B\}$.

We use three different metrics in order to evaluate the solutions quality:

1. First, for each exact and approximate solution, we calculate the cardinalities of the generated Pareto fronts. We denote by $|\mathcal{S}^*|$ the cardinality of the exact Pareto front $\mathcal{S}^*$, and by $|\mathcal{S}|$ the cardinality of the near Pareto front $|\mathcal{S}|$.
2. Using the cardinalities, we define $\%S$, this metric calculates the number of exact solutions generated by each heuristic, it is given by $|\mathcal{S} \cap \mathcal{S}^*|/|\mathcal{S}|$.

3. GD is a generational distance, it calculates the average of the minimum Euclidian distances between the approximate solutions and the exact ones. It is given by $GD = \frac{1}{|\mathcal{S}|}\left(\sum_{s_1 \in \mathcal{S}} \min_{s_2 \in \mathcal{S}^*} d_{s_1,s_2}\right)$ where d_{s_1,s_2} is the Euclidian distance between the element $s_1 \in \mathcal{S}$ and the element $s_2 \in \mathcal{S}^*$.
4. $\mathcal{H}$ is the *Hypervolume*, it calculates the area dominated by some front. In the following, we give the ratio of the area dominated by the approach pareto front and not by the exact Pareto front. Even when GD are small, they may be a poor indicator of the quality of the front $\mathcal{S}$. Therefore we introduced the metric $\mathcal{H}$ in order to estimate the repartition of the approach solutions on the exact Pareto front (see Fig. 2).
5. Furthermore, we calculate the CPU running time needed for each method.

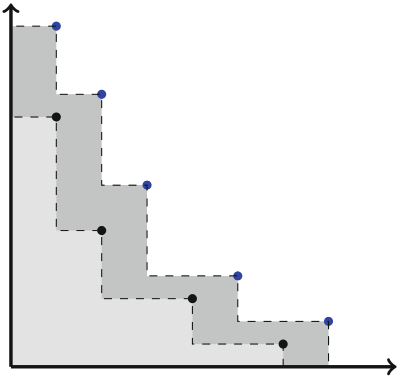


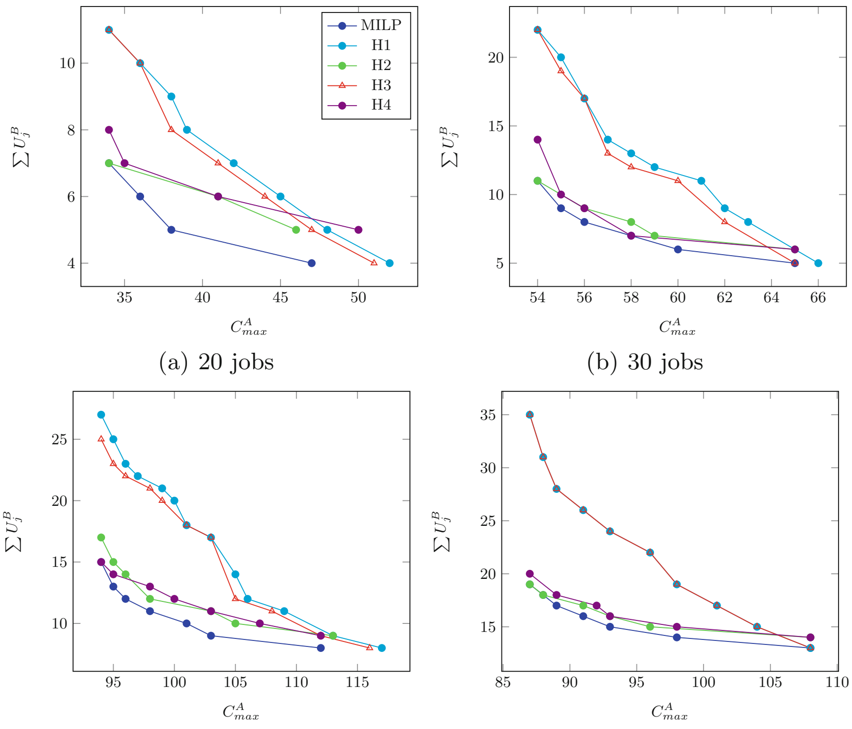
Fig. 2. Hypervolume measures representations

In the following, we first analyze the results graphically in the objective space (see Fig. 3) by considering four representative instances of obtained solutions. We also report complete results in Tables 1, 2 and 3 with some discussions.

7.1 Graphical Representation for Four Instances

In this section, we provide some graphical examples of the Pareto fronts generated by the exact and the heuristics methods. The chosen instances are fairly representative of the other instances. We can see in Figs. 3(a), (b), (c) and (d) five curves, such that the blue one gives the exact Pareto front, where the others are resulting from the heuristics. From this graphical, we can easily see that the green and purple curves are better that the others. They are the results of the heuristics 2 and 4.

In the next section we analyze and discuss all the results and give the average of the metrics previously introduced.



(a) 20 jobs

(b) 30 jobs

(c) 40 jobs

(d) 50 jobs

Fig. 3. Example of the obtained Pareto fronts with four instances with 20, 30, 40 and 50 jobs. (Color figure online)

Table 1. Comparison of the performances of the MILPs

	MILP-Time		*MILP-Assign*	
n	*CPU*	$\lvert\mathcal{S}^*\rvert$	CPU	$\lvert\mathcal{S}\rvert$
10	0.01	2.37	1.281	2.37
20	0.69	4.07	708.39	4.07
30	2.65	4.87	-	-
40	12.20	6.13	-	-
50	78.00	7.50	-	-
70	4664.16	9.766		

7.2 Result Tables and Discussions

In this section, we present the quantitative comparison results.

Table 1 gives the performances to the proposed mathematical formulations (MILP-Time and MILP-Assign).

Table 2. Performance comparison using the CPU and $|\mathcal{S}|$

	H1		H2		H3		H4	
n	CPU	$\|\mathcal{S}\|$	CPU	$\|\mathcal{S}\|$	CPU	$\|\mathcal{S}\|$	CPU	$\|\mathcal{S}\|$
10	0.00	2.87	0.00	2.43	0.000	2.97	0.000	2.53
20	0.00	5.63	0.00	4.07	0.000	5.40	0.000	4.03
30	0.00	7.50	0.00	4.90	0.001	7.13	0.000	4.87
40	0.00	9.67	0.00	6.20	0.002	9.30	0.001	5.97
50	0.00	11.53	0.00	7.27	0.006	11.40	0.003	6.77
70	0.00	15.23	0.00	9.60	0.019	15.13	0.005	8.63

Table 3. Performance comparison using the $\%S$, GD and $\mathcal{H}$

	H1			H2			H3			H4		
n	$\%S$	GD	$\mathcal{H}$	$\%S$	GD	$\mathcal{H}$	$\%S$	GD	$\mathcal{H}$	$\%S$	GD	$\mathcal{H}$
10	29.83	0.86	24.21	36.94	0.68	17.90	33.28	0.82	25.66	28.33	0.89	21.11
20	14.40	1.39	37.39	28.85	0.91	11.52	18.85	1.39	36.31	12.08	1.24	19.95
30	6.09	1.86	40.94	25.19	1.06	10.37	7.08	1.89	40.54	9.82	1.44	21.23
40	1.78	2.02	41.13	27.04	1.12	7.86	1.84	2.06	41.58	9.66	1.43	20.07
50	1.01	2.47	44.52	37.08	1.01	5.65	1.21	2.48	44.50	14.50	1.53	17.69
70	0.70	3.09	45.30	35.74	0.96	4.77	0.70	3.10	45.29	14.13	1.39	10.51

About MILPs performances, we can easily see that the time indexed formulation is better than the assigned formulation, since its solves instances with 70 jobs in 1 h and 18 min on average, the average time spent to obtain the whole Pareto front. Unfortunately, the assigned indexed formulation shows weaker results since its cannot solve instances with more than 20 jobs. We note that the time indexed formulation has a pseudo-polynomial binary variables, so the present performances are due to the small values of the processing times.

Table 2 represents the CPU average and the cardinality $|\mathcal{S}|$ of each heuristic.

Table 3 is dedicated to the comparison using the metrics $\%S$, GD and $\mathcal{H}$.

Table 1 shows that the number of compromise solutions increases with the increase of the number of jobs. This is also true for the heuristics (see Table 2), where more compromise solutions are found using the heuristics 1 and 3. However, this does not necessarily mean that these solutions are better than the solutions found by the heuristics 2 and 4. In fact, we have seen in Fig. 3, that more solutions are proposed by H1 and H3, but the corresponding curves are farther from the exact curve.

Although the criteria values are not normalized, the generational distances are still small for heuristics 2 and 4. In fact, for the instances with 70 jobs, solutions given by H2 are in average within distance 0.92 from the exacts Pareto solutions. And for the same instances solutions given by H4 are in average within distance 1.39 from the exact Pareto solutions. In the case of H1 and H3, the distances are slightly more significant. In fact, the values reach 3 for the instances with 70 jobs.

The generational distance values show that the compromise solutions proposed by the heuristics are close to the exact Pareto solution obtained by the MILPs. However, when analyzing the multi-criteria heuristic, we need to know whether the near Pareto solutions are well distributed in the criteria space or not. The *Hypervolume* can be a good gauge to measure the repartition of the solutions obtained by heuristics. Table 3, shows that H2 is the best method according to metric $\mathcal{H}$. Surprisingly, the values of H2 and H4 are decreasing with the increasing of the number of jobs.

8 Conclusion and Perspectives

This paper tackled multi-agent scheduling problem, which is featured by two non-disjoint agents A and B, competing to perform their jobs on two identical parallel machines. Agent A aims at minimizing the maximum completion times of its jobs, whereas agent B tries to minimize the number of its tardy jobs.

The studied problem result from the application of the well known ε-constraint approach. This mono-criteria problem minimizes agent A makespan under a bounded constraint on agent B objective function. We proposed two types of mathematical programming formulation. The first one, is based on precedence decision variables, while the other is based on time indexing decision variables. The empirical results showed that when the processing times were in $[1, 10]$ with two identical machines, the time indexed formulation allows to obtain Pareto front for large instances.

In the second part of this research we proposed four heuristics for this NP-hard problem. The first and second heuristics are polynomial, they are based on LPT-FAM order. The third and fourth heuristics are inspired by the first two ones, instead of using an approach method for minimizing the makespan, it uses an exact algorithm based on dynamic programming. These heuristics are finally compared with the MILPs, using generational distance, hypervolume and cardinality metrics, the results showed that the heuristics 2 and 4 give better solutions. All proposed methods can be used to solve the case of m identical parallel machines. Thereby, we have conducted some other experimental results with 4 and 6 identical parallel machines and not presented here. In this case, the preliminary obtained results display the same conclusions.

For further research, we will propose a genetic algorithm starting from the solutions obtained by the heuristics. It would be also interesting to seek for a pseudo-polynomial time algorithm.

Acknowledgments. This research was partially funded by the European Union through the Erasmus STA program.

References

1. Agnetis, A., Mirchandani, P., Pacciarelli, D., Pacifici, A.: Non-dominated schedules for a job-shop with two competing users. Comput. Math. Organ. Theory **6**(2), 191–217 (2000)
2. Agnetis, A., Mirchandani, P., Pacciarelli, D., Pacifici, A.: Scheduling problems with two competing agents. Oper. Res. **52**, 229–242 (2004)
3. Agnetis, A., Billaut, J.-C., Gawiejnowicz, S., Pacciarelli, D., Soukhal, A.: Multiagent Scheduling Models and Algorithms. Springer, Heidelberg (2014). 271 p
4. Baker, K.R., Smith, J.C.: A multiple-criteria model for machine scheduling. J. Sched. **6**, 7–16 (2003)
5. Balasubramanian, H., Fowler, J., Keha, A., Pfund, M.: Scheduling interfering job sets on parallel machines. Eur. J. Oper. Res. **199**, 55–67 (2009)
6. Blazewicz, J., Ecker, K.H., Pesch, E., Schmidt, G., Weglarz, J.: Handbook on Scheduling: From Theory to Applications. International Handbooks on Information Systems. Springer, Heidelberg (2007)
7. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**, 182–197 (2002)
8. Huynh Tuong, N., Soukhal, A., Billaut, J.C.: Single-machine multi-agent scheduling problems with a global objective function. J. Sched. **15**(3), 311–321 (2012)
9. Kung, H.T., Luccio, F., Preparata, F.P.: On finding the maxima of a set of vectors. J. Associ. Comput. Mach. **22**(4), 469–476 (1975)
10. Ng, C.T., Cheng, T.C.E., Yuan, J.J.: A note on the complexity of the problem of two-agent scheduling on a single machine. J. Comb. Optim. **12**(4), 387–394 (2006)
11. Peha, J.M.: Heterogeneous-criteria scheduling: minimizing weighted number of tardy jobs and weighted completion time. Comput. Oper. Res. **22**(10), 1089–1100 (1995)
12. Sadi, F., Soukhal, A., Billaut, J.-C.: Solving multi-agent scheduling problems on parallel machines with a global objective function. RAIRO - Oper. Res. **48**(2), 225–269 (2014)

An Evaluative Model to Assess the Organizational Efficiency in Training Corporations

Ana Fernandes[1], Henrique Vicente[2,3], Margarida Figueiredo[2,4], Mariana Neves[5], and José Neves[3](✉)

[1] Organização Multinacional de Formação, Lisbon, Portugal
anavilafernandes@gmail.com
[2] Departamento de Química, Escola de Ciências e Tecnologia, Universidade de Évora, Évora, Portugal
{hvicente,mtf}@uevora.pt
[3] Centro Algoritmi, Universidade do Minho, Braga, Portugal
jneves@di.uminho.pt
[4] Centro de Investigação em Educação e Psicologia, Universidade de Évora, Évora, Portugal
[5] Deloitte, London, UK
maneves@deloitte.co.uk

Abstract. In an organisation any optimization process of its issues faces increasing challenges and requires new approaches to the organizational phenomenon. Indeed, in this work it is addressed the problematic of efficiency dynamics through intangible variables that may support a different view of the corporations. It focuses on the challenges that information management and the incorporation of context brings to competitiveness. Thus, in this work it is presented the analysis and development of an intelligent decision support system in terms of a formal agenda built on a Logic Programming based methodology to problem solving, complemented with an attitude to computing grounded on Artificial Neural Networks. The proposed model is in itself fairly precise, with an overall accuracy, sensitivity and specificity with values higher than 90 %. The proposed solution is indeed unique, catering for the explicit treatment of incomplete, unknown, or even self-contradictory information, either in a quantitative or qualitative arrangement.

Keywords: Optimization · Efficiency · Logic programming · Knowledge representation · Artificial neural networks

1 Introduction

The *Excellence Model* (*EM*), proposed by the *European Foundation for Quality Management* (*EFQM*), is a self-assessment framework to measure the strengths and areas of improvement of an organisation. The term *Excellence* is used once the *EM* focuses on what an organisation does, or could do, in order to provide an excellent service or product to its customers, users or stakeholders [1, 2]. While its origins

T.K. Dang et al. (Eds.): FDSE 2016, LNCS 10018, pp. 415–428, 2016.
DOI: 10.1007/978-3-319-48057-2_29

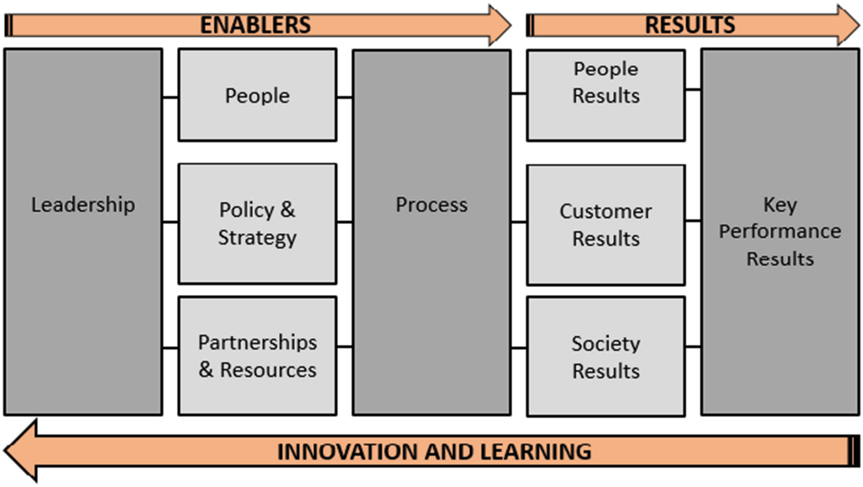


Fig. 1. The *Excellence Model* framework (adapted from http://www.efqm.org/)

remain in the private sector, public and voluntary sector organisations can also benefit from using the *EM*. It is non-prescriptive and does not involve a strict respect of a set of rules or standards, but provides a broad and coherent set of assumptions about what is required for an efficient organisation and its management [1]. The *EM* starts with the premise epitomized in Fig. 1, i.e., *People*, *Customer* and *Society Results* achieved through *Leadership* driving *People*, *Policy and Strategy*, *Partnerships and Resources,* leading ultimately to excellence in *Key Performance Results* [1–3]. Indeed, there are nine main ideas or criteria in the *EM* that underpin this premise and attempt to cover all organisations' activities, namely:

- Leadership;
- People;
- Policy and Strategy;
- Partnerships and Resources;
- Processes;
- People Results;
- Customer Results;
- Society Results; and
- Key Performance Results.

These nine criteria are separated into *Enablers* (the former five issues) and *Results* (the remaining ones). The former ones are concerned with how the organisation conducts itself, how it manages its staff and resources, how it plans the strategy and how it reviews and monitors key processes. The *Results* are what it achieves and encompasses the level of satisfaction among the organisations' employees and customers, its impact on the community and the values of key performance indicators [1–3].

Each one of these criteria is subdivided in order to describe in more detail the concept of *Excellence* in a specific area, and to auto-examine the performance of the organisation.

The starting point for most organisations is to gather relevant evidence to the criteria of the *EM*. This involves asking, for each criterion, *How good are we*? And *How could we improve*?. Thus, the *Evidence* can take a variety of forms depending on the organisation [1, 4], and it is mandatory that each organisation has to look at the framework that suits it better [5].

The present study address the theme of the *Organizational Efficiency* and the focus was put on the criterion *People Results*, in particular with regard to employees' satisfaction. However, employees' satisfaction is a complex phenomenon that involves a large number of factors, some of which depend on the worker in itself, and on the organisation [3, 5, 6]. Consequently, it is difficult to assess the *Organizational Efficiency* since it needs to consider different conditions with complex relations among them, where the available data may be incomplete/unknown (e.g., absence of answers to some questions presented in the questionnaire), and/or contradictory (e.g., questions relating to the same issue with incongruous answers). In order to overcome these difficulties, the present work reports the founding of an intelligent computational framework that uses *Logic Prog*ramming based techniques to knowledge representation in order to set the structure of the information and the associate inference mechanisms [7], i.e., it will be centered on a *Proof Theoretical* approach to problem solving, complemented with a computational framework based on *Artificial Neural Networks* (*ANNs*), selected due to their dynamics like adaptability, robustness, and flexibility [8].

2 Knowledge Representation

Many approaches to knowledge representation have been proposed using the *Logic Programming* (*LP*) epitome, namely in the area of *Model Theory* [9, 10], and *Proof Theory* [7, 11]. In the present work the *Proof Theoretical* approach in terms of an extension to the *LP* language is followed, where a logic program is a finite set of clauses, given in the form.

$$
\begin{array}{l}
\{ \\
\quad \neg p \leftarrow not\, p, not\, exception_p \\
\quad p \leftarrow p_1, \cdots, p_n, not\, q_1, \cdots, not\, q_m \\
\quad ?(p_1, \cdots, p_n, not\, q_1, \cdots, not\, q_m)(n, m \geq 0) \\
\quad exception_{p_1} \\
\quad \ldots \\
\quad exception_{p_j}\ (0 \leq j \leq k), being\, k\, and\, integer\, number \\
\}\ :: scoring_{value}
\end{array}
$$

where the first clause stand for predicate's closure, "," denotes "*logical and*", while "*?*" is a domain atom denoting falsity. The p_i, q_j, and p are classical ground literals, i.e., either positive atoms or atoms preceded by the classical negation sign ¬ [7]. Indeed, ¬ stands for a strong declaration that speaks for itself, and *not* denotes *negation-by-failure*, or in other words, a flop in proving a given statement, once it was not declared explicitly. Under symbols' theory, every program is associated with a set of abducibles

[9, 10], given here in the form of exceptions to the extensions of the predicates that make the program, i.e., clauses of the form:

$$exception_{p_1}, \cdots, exception_{p_j}\ (0 \leq j \leq k), being\ k\ an\ integer\ number$$

that stand for data, information or knowledge that cannot be ruled out. On the other hand, clauses of the type:

$$?(p_1, \cdots, p_n, not\, q_1, \cdots, not\, q_m)\ (n, m \geq 0)$$

also named invariants or restrictions, allows one to set the context under which the universe of discourse has to be understood. The term $scoring_{value}$ stands for the relative weight of the extension of a specific *predicate* with respect to the extensions of peers ones that make the inclusive or global program.

2.1 Quantitative Knowledge

In order to set one's approach to knowledge representation, two metrics will be set, namely the Quality-of-Information (*QoI*) of a logic program that will be understood as a mathematical function that will return a truth-value ranging between 0 and 1 [12, 13], once it is fed with the extension of a given predicate, i.e., QoI_{i} = 1 when the information is *known* (*positive*) or *false* (*negative*) and QoI_i = 0 if the information is *unknown*. For situations where the extensions of the predicates that make the program also include *abducible* sets, its terms (or clauses) present a $QoI_i \in [0, 1]$, in the form:

$$QoI_i = {}^{1}\!/_{Card} \tag{1}$$

if the *abducible* set for *predicates i* and *j* satisfy the *invariant*:

$$?\big(\big(exception_{p_i}; exception_{p_j}\big), \neg\big(exception_{p_i}; exception_{p_j}\big)\big)$$

where ";" denotes "*logical or*" and "*Card*" stands for set cardinality, being $i \neq j$ and $i, j \geq 1$ (a pictorial view of this process is given in Fig. 2(a), as a pie chart).

On the other hand, the clauses cardinality (*K*) will be given by $C_1^{Card} + \cdots + C_{Card}^{Card}$, if there is no constraint on the possible combinations among the abducible clauses, being the *QoI* acknowledged as:

$$QoI_{i_{1 \leq i \leq Card}} = {}^{1}\!/_{C_1^{Card}}, \cdots, {}^{1}\!/_{C_{Card}^{Card}} \tag{2}$$

where C_{Card}^{Card} is a card-combination subset, with *Card* elements. A pictorial view of this process is given in Fig. 2(b), as a pie chart.

However, a term's *QoI* also depends on their attribute's *QoI*. In order to evaluate this metric, look to Fig. 3, where the segment with bounds 0 and 1 stands for every attribute domain, i.e., all the attributes range in the interval [0, 1]. [*A*, *B*] denotes the range where the unknown attributes values for a given predicate may occur (Fig. 3).

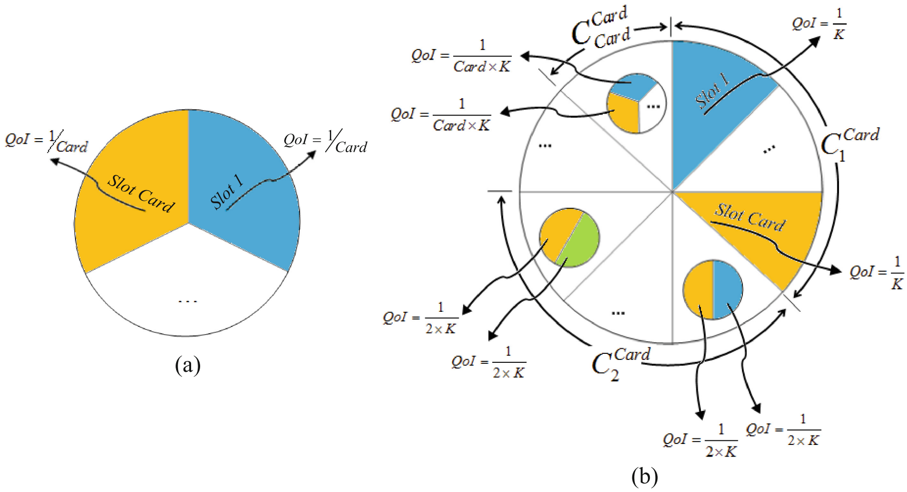


Fig. 2. QoI's values for the abducible set for *predicate_i* with (a) and without (b) constraints on the possible combinations among the abducible clauses

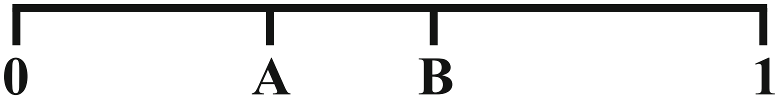


Fig. 3. Setting the *QoIs* of each attribute's clause

$$QoI_{attribute_i} = 1 - ||A - B|| \qquad (3)$$

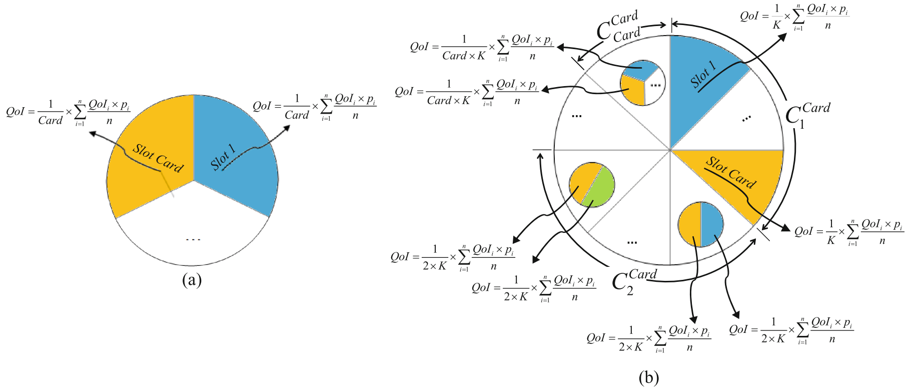
where ||A − B|| stands for the modulus of the arithmetic difference between *A* and *B*. Therefore, in Fig. 4 is showed the *QoI's* values for the abducible set for $predicate_i$.

Under this setting, another metric has to be considered, which will be denoted as *DoC* (*Degree-of-Confidence*), that stands for one's confidence that the argument values or attributes of the terms that make the extension of a given predicate, having into consideration their domains, are in a given interval [14]. The *DoC* is figured using $DoC = \sqrt{1 - \Delta l^2}$, where Δl stands for the argument interval length, which was set to the interval [0, 1] (Fig. 5).

Thus, the universe of discourse is engendered according to the information presented in the extensions of such predicates, according to productions of the type:

$$\begin{aligned} predicate_i - \bigcup_{1 \le j \le m} clause_j(((A_{x_1}, B_{x_1})(QoI_{x_1}, DoC_{x_1})), \cdots, \\ ((A_{x_l}, B_{x_l})(QoI_{x_l}, DoC_{x_l}))) :: QoI_j :: DoC_j \end{aligned} \qquad (4)$$

where ∪, *m* and *l* stand, respectively, for *set union*, the *cardinality* of the extension of $predicate_i$ and the number of attributes of each clause [14]. On the other hand, either



$QoI = \frac{1}{Card} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
$QoI = \frac{1}{Card} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
Slot Card
Slot 1
...
(a)
C_{Card}^{Card}
$QoI = \frac{1}{Card \times K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
$QoI = \frac{1}{Card \times K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
$QoI = \frac{1}{K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
Slot 1
C_1^{Card}
Slot Card
$QoI = \frac{1}{K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
$QoI = \frac{1}{2 \times K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
$QoI = \frac{1}{2 \times K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
C_2^{Card}
$QoI = \frac{1}{2 \times K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
$QoI = \frac{1}{2 \times K} \times \sum_{i=1}^{n} \frac{QoI_i \times p_i}{n}$
...
(b)

Fig. 4. *QoI's* values for the abducible set for *predicate*$_i$ with (a) and without (b) constraints on the possible combinations among the abducible clauses. $\sum_{i=1}^{n} (QoI_i \times p_i)/n$ denotes the *QoI's* average of the attributes of each clause (or term) that sets the extension of the predicate under analysis. n and p_i stand for, respectively, for the attribute's cardinality and the relative weight of attribute p_i with respect to its peers $(\sum_{i=1}^{n} p_i = 1)$

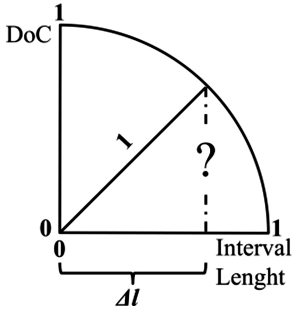


Fig. 5. Evaluation of the attributes' degree of confidence

the subscripts of the QoI_s and the DoC_s, or those of the pairs (A_s, B_s), i.e., x_1, ..., x_l, stand for the attributes' clauses values ranges.

2.2 Qualitative Knowledge

In present study both qualitative and quantitative data/knowledge are present. Aiming at the quantification of the qualitative part and in order to make easy the understanding of the process, it will be presented in a graphical form. Taking as an example a set of n issues regarding a particular subject (where the criteria are *zero*, *low*, ..., *high* and *very high*), let us itemized an unitary area circle split into n slices (Fig. 5). The marks in the axis correspond to each of the possible options. If the answer to issue 1 is *high* the correspondent area is $\pi \times \left(\sqrt{\frac{0.75}{\pi}}\right)^2/n$, i.e., $0.75/n$ (Fig. 6(a)). Assuming that in the

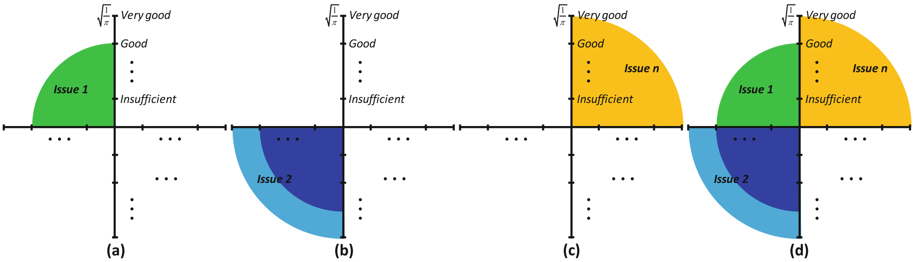


Fig. 6. A view of a qualitative data/knowledge processing

issue 2 are chosen the alternatives *high* and *very high*, the correspondent area ranges between $\left[\pi \times \left(\sqrt{\frac{0.75}{\pi}}\right)^2/n, \pi \times \left(\sqrt{\frac{1}{\pi}}\right)^2/n\right]$, i.e., $[0.75/n, 1/n]$ (Fig. 6(b)). Finally, in issue n if no alternative is ticked, all the hypotheses should be considered and the area varies in the interval $\left[0, \pi \times \left(\sqrt{\frac{1}{\pi}}\right)^2/n\right]$, i.e., $[0, 1/n]$ (Fig. 6(c)). The total area is the sum of the partial ones (Fig. 6(d)).

3 Methods

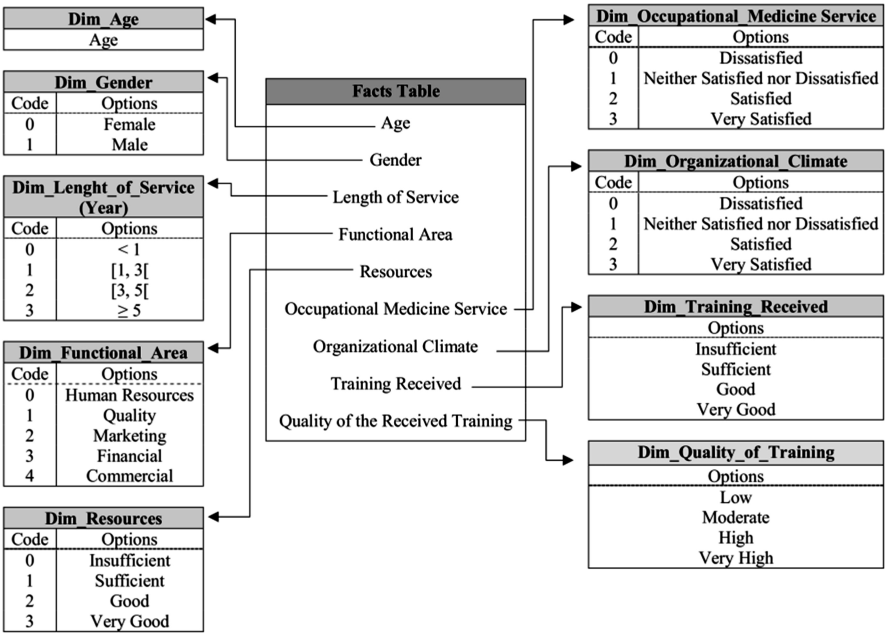
Aiming to develop a predictive model to assess employees' satisfaction a questionnaire was set and applied to a cohort of 78 workers of training companies. This section describes briefly the data collection tool and how the information is processed.

3.1 Questionnaire

The questions included in the questionnaire aimed to evaluate the degree of employee satisfaction about the performance of the organisation and identify areas of strength and areas for improvement, in order to trace new goals. The respondents participated in the study voluntarily and the questionnaires were anonymous to ensure the confidentiality of information provided. The questions included in the questionnaire were organized into two sections, where the former one includes the questions related with employees' age, gender, length of service and functional area. The last one comprises questions related with the employees' opinions about the resources, occupational medicine service, organizational climate, training and quality of training. In Fig. 7 shows the possible answers to each question (*Option* column) and how the responses were codified (*Code* column).

3.2 Extract, Transform and Load

To develop a predictive model it was necessary to gather data from several sources and carry out an *extract*, *transform* and *load* process to organize the information.



Dim_Age
Age

Dim_Gender	
Code	Options
0	Female
1	Male

Dim_Lenght_of_Service (Year)	
Code	Options
0	< 1
1	[1, 3[
2	[3, 5[
3	≥ 5

Dim_Functional_Area	
Code	Options
0	Human Resources
1	Quality
2	Marketing
3	Financial
4	Commercial

Dim_Resources	
Code	Options
0	Insufficient
1	Sufficient
2	Good
3	Very Good

Facts Table
Age
Gender
Length of Service
Functional Area
Resources
Occupational Medicine Service
Organizational Climate
Training Received
Quality of the Received Training

Dim_Occupational_Medicine Service	
Code	Options
0	Dissatisfied
1	Neither Satisfied nor Dissatisfied
2	Satisfied
3	Very Satisfied

Dim_Organizational_Climate	
Code	Options
0	Dissatisfied
1	Neither Satisfied nor Dissatisfied
2	Satisfied
3	Very Satisfied

Dim_Training_Received
Options
Insufficient
Sufficient
Good
Very Good

Dim_Quality_of_Training
Options
Low
Moderate
High
Very High

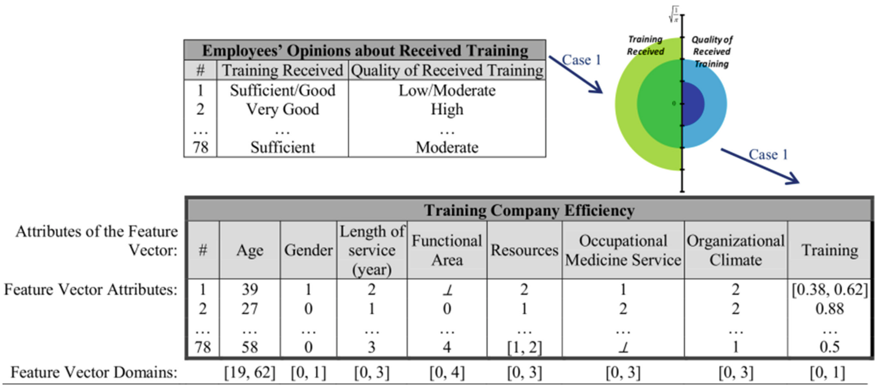
Fig. 7. An overview of the data model

The information was organized in a *star schema*, which consists of a collection of tables that are logically related to each other [15]. To obtain a star schema it was essential to follow a few steps. In the former one it was necessary to understand the problem and gather the parameters data, information or knowledge that may have influence in the final outcome. Based on literature [4–6], the parameters that may influence the satisfaction of workers in a company were age, gender, length of service, functional area, resources, satisfaction with occupational medicine service, organizational climate, received training and quality of the received training. The following stage was related with the dimensions that would be needed to define these parameters on the facts table. Finally, information from several sources was collected, transformed according the fact and dimension table and loaded to fact table.

The star schema conceived for this study (Fig. 7) takes into account the variables that influence the satisfaction of workers (Facts Table) where Dim Tables show how data was classified. For example, regarding length of service, employees with less than a year, comprised in the range [1, 3], ranging between [3, 5], and with more than 5 years, were epitomize by the values 0 (zero), 1 (one), 2 (two) or 3 (three), respectively.

3.3 A Logic Programming Approach to Data Processing

Based on the star schema presented in Fig. 7, it is possible to build up a knowledge database given in terms of the table depicted in Fig. 8, which stand for a situation



Employees' Opinions about Received Training		
#	Training Received	Quality of Received Training
1	Sufficient/Good	Low/Moderate
2	Very Good	High
…	…	…
78	Sufficient	Moderate

	Training Company Efficiency								
Attributes of the Feature Vector:	#	Age	Gender	Length of service (year)	Functional Area	Resources	Occupational Medicine Service	Organizational Climate	Training
Feature Vector Attributes:	1	39	1	2	⊥	2	1	2	[0.38, 0.62]
	2	27	0	1	0	1	2	2	0.88
	…	…	…	…	…	…	…	…	…
	78	58	0	3	4	[1, 2]	⊥	1	0.5
Feature Vector Domains:		[19, 62]	[0, 1]	[0, 3]	[0, 4]	[0, 3]	[0, 3]	[0, 3]	[0, 1]

Fig. 8. A fragment of the knowledge base for training company efficiency evaluation

where one has to manage information aiming to in order to estimate the training company efficiency. Under this scenario some incomplete and/or unknown data is also available. For instance, in the former case the *Functional Area* is unknown (depicted by the symbol ⊥), while the opinion about *Training Received* is not conclusive (*Sufficient/Good*).

Applying the algorithm presented in [14] to the table or relation's fields that make the knowledge base for training company efficiency assessment (Fig. 8), and looking to the *DoCs* values obtained as described before, it is possible to set the arguments of the predicate ***effic**iency* (*effic*) referred to below, whose extensions denote the objective function with respect to the problem under analyze:

$$effic : Age,\ Gen_{der},\ L_{ength}O_{f}S_{ervice},\ F_{unctional}A_{rea},\ Res_{ources},$$

$$O_{ccupational}M_{edicine}S_{ervice},\ O_{rganizational}C_{limate},\ T_{raining} \rightarrow \{0, 1\}$$

where 0 (zero) and 1 (one) denote, respectively, the truth values *false* and *true*.

The algorithm presented in [14] encompasses different phases. In the former one the clauses or terms that make extension of the predicate under study are established. In a second step the boundaries of the attributes intervals are set in the interval [0, 1] according to a normalization process given by the expression $(Y - Y_{min})/(Y_{max} - Y_{min})$, where the Y_s stand for themselves. Finally, the *DoC* is evaluated as described in Sect. 2.1.

Exemplifying the application of the algorithm presented in [14], to a term (employee) that presents the feature vector ($Age = 48$, $Gen_{der} = 1$, $L_{ength\ of}\ S_{ervice} = 3$, $F_{unctional}\ A_{rea} = \perp$, $Res_{ources} = 1$, $O_{ccupational}\ M_{edicine}\ S_{ervice} = 2$, $O_{rganizational}\ C_{limate} = 1$, $T_{raining} = [0.5, 0.62]$), one may have:

Begin (DoCs evaluation)

The predicate's extension that sets the Universe-of-Discourse to the case (term) under observation is fixed

$$
\begin{array}{l}
\{ \\
\neg \mathit{effic}\left(\left(\left(A_{Age}, B_{Age}\right)\left(QoI_{Age}, DoC_{Age}\right)\right), \cdots, \left((A_T, B_T)(QoI_T, DoC_T)\right)\right) \\
\quad \leftarrow \mathit{not}\ \ \mathit{effic}\left(\left(\left(A_{Age}, B_{Age}\right)\left(QoI_{Age}, DoC_{Age}\right)\right), \cdots, \left((A_T, B_T)(QoI_T, DoC_T)\right)\right) \\
\mathit{effic}\left(\left((48,\ 48)(QoI_{48}, DoC_{48})\right), \cdots, \left((0.5, 0.62)\left(QoI_{[0.5,0.62]}, DoC_{[0.5,0.62]}\right)\right)\right) \\
\qquad \underbrace{[19, 62] \qquad \ldots \qquad\qquad [0,\ 1]}_{attribute's\ \ domain} \\
\}\ :: 1 :: DoC
\end{array}
$$

The attribute's boundaries are set to the interval [0, 1], according to a normalization process that uses the expression $(Y - Y_{min})/(Y_{max} - Y_{min})$

$$
\begin{array}{l}
\{ \\
\neg \mathit{effic}\left(\left(\left(A_{Age}, B_{Age}\right)\left(QoI_{Age}, DoC_{Age}\right)\right), \cdots, \left((A_T, B_T)(QoI_T, DoC_T)\right)\right) \\
\quad \leftarrow \mathit{noteffic}\left(\left(\left(A_{Age}, B_{Age}\right)\left(QoI_{Age}, DoC_{Age}\right)\right), \cdots, \left((A_T, B_T)(QoI_T, DoC_T)\right)\right) \\
\mathit{effic}\left(\left((0.67, 0.67)(1_{0.67}, DoC_{0.67})\right), \cdots, \left((0.5, 0.62)\left(1_{[0.5,0.62]}, DoC_{[0.5,0.62]}\right)\right)\right) \\
\hfill :: 1 :: DoC \\
\qquad \underbrace{[0,\ 1] \qquad\quad \ldots \qquad\quad [0,\ 1]}_{attribute`s\ \ domains\ \ once\ \ normalized}
\end{array}
$$

The DoC's values are evaluated

$$
\begin{array}{l}
\{ \\
\neg \mathit{effic}\left(\left(\left(A_{Age}, B_{Age}\right)\left(QoI_{Age}, DoC_{Age}\right)\right), \cdots, \left((A_T, B_T)(QoI_T, DoC_T)\right)\right) \\
\quad \leftarrow \mathit{noteffic}\left(\left(\left(A_{Age}, B_{Age}\right)\left(QoI_{Age}, DoC_{Age}\right)\right), \cdots, \left((A_T, B_T)(QoI_T, DoC_T)\right)\right) \\
\mathit{effic}(((0.67, 0.67)(1, 1)), \cdots, ((0.5, 0.62)(1, 0.99))) :: 1 :: 0.87 \\
\qquad \underbrace{[0,\ 1] \qquad\quad \ldots \quad [0,\ 1]}_{attribute`s\ \ domains\ \ once\ \ normalized} \\
\}\ :: 1
\end{array}
$$

End

4 Artificial Neural Networks

The framework presented previously shows how the information comes together and how it is processed. In this section, a data mining approach to deal with the processed information is considered. A hybrid computing approach was set to model the universe of discourse, where the computational part is based on *ANNs*, which are used not only to structure data but also to capture the objective function's nature (i.e., the relationships between inputs and outputs) [16, 17].

ANNs denote a set of connectionist models inspired in the behaviour of the human brain. In particular, the *Multilayer Perceptron* (*MLP*) is the most popular *ANN* architecture, where neurons are grouped in layers and only forward connections exist [18]. This provides a powerful base-learner, with advantages such as nonlinear mapping and noise tolerance, increasingly used in data mining due to its good behaviour in terms of predictive knowledge. *MLP* is molded on three or more layers of artificial neurons, including an input layer, an output layer and a number of hidden layers with a certain number of active neurons with connections with adjustable weights. In addition, there is also a bias, which is only connected to neurons in the hidden and output layers. The number of nodes in the input layer sets the number of independent variables, and the number of nodes in output layer denotes the number of dependent ones [18].

Figure 9 shows a case being submitted to the organizational efficiency assessment. The normalized values of the interval boundaries and its *QoI's* and *DoC's* stand for the inputs to the *ANN*. The output is given in terms of organizational efficiency evaluation and the degree of confidence that one has on such a happening. In this study 78 responses to the questionnaire were considered (i.e., seventy eight terms or clauses of the extension of predicate *effic*). To implement the evaluation mechanisms and to test the model, ten folds cross validation were applied [18]. The back propagation algorithm was used in the learning process of the *MLP*. As the output function in the pre-processing layer it was used the identity one, while in the other layers we considered the sigmoid function.

A common tool to evaluate the results presented by the classification models is the coincidence matrix, a matrix of size $L \times L$, where L denotes the number of possible

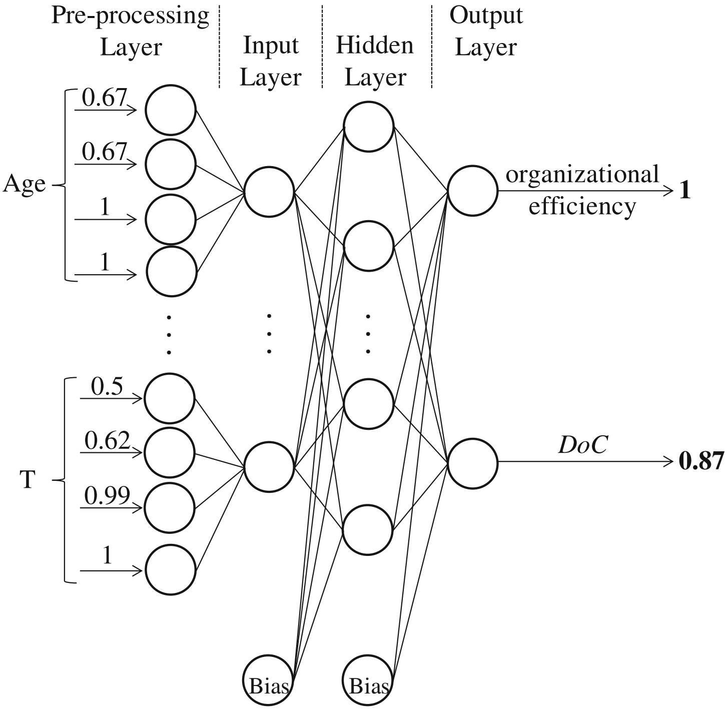


Fig. 9. The *ANN* topology

Table 1. The coincidence matrix for the *ANN* model

Target	Predictive	
	True (1)	False (0)
True (1)	54	4
False (0)	2	18

classes. This matrix is created by matching the predicted and target values. *L* was set to 2 (two) in the present case. Table 1 presents the coincidence matrix of the *ANN* model, where the values presented denote the average of 30 (thirty) experiments. A glance at Table 1 shows that the model accuracy was 92.3 % (72 instances correctly classified in 78). Therefore, the predictions made by the *ANN* model are satisfactory, attaining accuracies higher than 90 %.

Based on coincidence matrix it is possible to compute sensitivity, specificity, *Positive Predictive Value* (*PPV*) and *Negative Predictive Value* (*NPV*) of the classifier. Briefly, sensitivity evaluates the proportion of true positives that are correctly identified as such, while specificity translates the proportion of true negatives that are correctly identified. *PPV* stands for the proportion of cases with positive results which are correctly classified, while *NPV* denotes the proportion of cases with negative results which are successfully labeled. The values obtained for sensitivity, specificity, *PPV* and *NPV* were 93.1 %, 90.0 %, 96.4 % and 81.8 %, respectively. Thus, it is our claim that the proposed model is able to evaluate the organizational efficiency properly, and can be a major contribution to achieve levels of excellence in highly competitive environments.

5 Conclusion

The organizational efficiency assessment is not only an inestimable practice, but something of utmost importance in the training context. The problems faced by contemporary society require from the training companies the highest standards of quality in the formation of future professionals. To meet this challenge it is necessary that the training companies optimize their efficiency in order to achieve excellence practices. However, it is difficult to assess the organizational efficiency since it is necessary to consider different variables and/or conditions, with complex relations entwined them, where the data may be incomplete, contradictory, and even unknown. In order to overcome these difficulties this work presents the founding of a hybrid computing approach that uses powerful knowledge representation techniques to set the structure of the information and the associate inference mechanisms, complemented with a computational framework based on *ANNs* (due to their proper dynamics, like adaptability, evolution, robustness, and flexibility). This approach not only allows evaluating the organizational efficiency but it also permits the estimation of the degree of confidence that one has on such a happening. In fact, this is one of the added values of this approach that arises from the complementary between *LP* (for knowledge representation) and the computing process based on *ANNs*.

Acknowledgments. This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

References

1. Vanagas, P., Mantas, V.: Development of total quality management in kaunas university of technology. Eng. Econ. **59**, 67–75 (2008)
2. Nabitz, U., Klazinga, N., Walburg, J.: The EFQM excellence model: European and Dutch experiences with the EFQM approach in health care. Int. J. Qual. Health Care **12**, 191–201 (2000)
3. Valk, P.: Quality assurance in postgraduate pathology training the Dutch way: regular assessment, monitoring of training programs but no end of training examination. Virchows Arch. **468**, 109–113 (2016)
4. Jianwei, Z., Yuxin, L.: Organizational climate and its effects on organizational variables: an empirical study. Int. J. Psychol. Stud. **2**, 190–201 (2010)
5. Dietz, D., Zwich, T.: The retention effect of training: portability, visibility and credibility. Discussion Paper No 16-011. http://ftp.zew.de/pub/zew-docs/dp/dp16011.pdf
6. Safanova, K., Podolskii, S.: Improvement of the evaluation of quality of the integrative intellectual resource of the higher educational establishment. Asian Soc. Sci. **11**, 112–124 (2015)
7. Neves, J.: A logic interpreter to handle time and negation in logic databases. In: Muller, R., Pottmyer, J. (eds.) Proceedings of the 1984 Annual Conference of the ACM on the 5th Generation Challenge, pp. 50–54. Association for Computing Machinery, New York (1984)
8. Cortez, P., Rocha, M., Neves, J.: Evolving time series forecasting ARMA models. J. Heuristics **10**, 415–429 (2004)
9. Kakas, A., Kowalski, R., Toni, F.: The role of abduction in logic programming. In: Gabbay, D., Hogger, C., Robinson, I. (eds.) Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 5, pp. 235–324. Oxford University Press, Oxford (1998)
10. Pereira, L., Anh, H.: Evolution prospection. In: Nakamatsu, K. (ed.) New Advances in Intelligent Decision Technologies – Results of the First KES International Symposium IDT 2009. Studies in Computational Intelligence, vol. 199, pp. 51–64. Springer, Berlin (2009)
11. Neves, J., Machado, J., Analide, C., Abelha, A., Brito, L.: The halt condition in genetic programming. In: Neves, J., Santos, M.F., Machado, J. (eds.) Progress in Artificial Intelligence. LNAI, vol. 4874, pp. 160–169. Springer, Berlin (2007)
12. Lucas, P.: Quality checking of medical guidelines through logical abduction. In: Coenen, F., Preece, A., Mackintosh, A. (eds.) Proceedings of AI-2003 (Research and Developments in Intelligent Systems XX), pp. 309–321. Springer, London (2003)
13. Machado, J., Abelha, A., Novais, P., Neves, J., Neves, J.: Quality of service in healthcare units. In: Bertelle, C., Ayesh, A. (eds.) Proceedings of the ESM 2008, pp. 291–298. Eurosis – ETI Publication, Ghent (2008)
14. Fernandes, F., Vicente, H., Abelha, A., Machado, J., Novais, P., Neves J.: Artificial neural networks in diabetes control. In: Proceedings of the 2015 Science and Information Conference (SAI 2015), pp. 362–370. IEEE Edition (2015)
15. O'Neil, P., O'Neil, B., Chen, X.: Star Schema Benchmark. Revision 3, 5 June 2009. http://www.cs.umb.edu/~poneil/StarSchemaB.pdf

16. Vicente, H., Couto, C., Machado, J., Abelha, A., Neves, J.: Prediction of water quality parameters in a reservoir using artificial neural networks. Int. J. Des. Nat. Ecodyn. **7**, 309–318 (2012)
17. Vicente, H., Dias, S., Fernandes, A., Abelha, A., Machado, J., Neves, J.: Prediction of the quality of public water supply using artificial neural networks. J. Water Supply: Res. Technol. – AQUA **61**, 446–459 (2012)
18. Haykin, S.: Neural Networks and Learning Machines. Pearson Education, New Jersey (2009)

Author Index