Editors-in-Chief

Prof. Hocine Cherif, Universite de Bourgogne, France
Dr. Rohaya Latip, University Putra Malaysia, Malaysia

Editorial Board

Ali Dehghan Tanha, University of Salford, United Kingdom
Ali Sher, American University of Ras Al Khaimah, UAE
Altaf Mukati, Bahria University, Pakistan
Andre Leon S. Gradwohl, State University of Campinas, Brazil
Azizah Abd Manaf, Universiti Teknologi Malaysia, Malaysia
Bestouen Ahmed, University Sains Malaysia, Malaysia
Carl Latino, Oklahoma State University, USA
Dariusz Jacek Jakóbczak, Technical University of Koszalin, Poland
Duc T. Pham, University of Birmingham, UK
E. George Dharma Prakash Raj, Bharathidasan University, India
Elboukhari Mohamed, University Mohamed First, Morocco
Eric Atwell, University of Leeds, United Kingdom
Eyas El-Qawasme, King Saud University, Saudi Arabia
Ezendu Arirwa, London Metropolitan University, United Kingdom
Fouzi Harrag, UFSAS University, Algeria
Genge Bela, University of Targu Mures, Romania
Guo Bin, Institute Telecom & Management SudParis, France
Hadj Hamma Tadjine, Technical University of Clausthal, Germany
Hassan Moradi, Qualcomm Inc., USA
Hocine Cherifi, Universite de Bourgogne, France
Isamou Shioya, Hosei University, Japan
Jacek Stando, Technical University of Lodz, Poland
Jan Piatos, VSB-Technical University of Ostrava, Czech Republic
Jose Filho, University of Grenoble, France
Juan Martinez, Gran Mariscal de Ayacucho University, Venezuela
Kaikai Xu, University of Electronic Science and Technology of China, China
Khaled A. Mahdi, Kuwait University, Kuwait
Ladislav Burita, University of Defence, Czech Republic
Mahtam Safar, Kuwait University, Kuwait
Majid Haghparast, Islamic Azad University, Shahre-Rey Branch, Iran
Martin J. Dudiak, Stratford University, USA
Mirel Cosulschi, University of Craiova, Romania
Monica Vladoiu, PG University of Ploiesti, Romania
Nan Zhang, George Washington University, USA
Noraziah Ahmad, Universiti Malaysia Pahang, Malaysia
Noradziah Ahmad, Universiti Malaysia Pahang, Malaysia
Pasquale De Meo, University of Applied Sciences of Porto, Italy
Paulino Leite da Silva, ISCAP-IPP University, Portugal
Piet Kommers, University of Twente, The Netherlands
Radhamani Govindaraju, Damodaran College of Science, India
Ramadhan Elaiess, University of Benghazi, Libya
Rasheed Al-Zahrni, King Saud University, Saudi Arabia
Salih Mohammad, University of Botswana, Botswana
Sutut Herawan, University Malaysia Pahang, Malaysia
Velayutham Pavanasm, Adhiparasakthi Engineering College, India
Viacheslav Wolfgangev, Lurinfo-MSU Institute, Russia
Waralak V. Srirachen, University of the Thai Chamber of Commerce, Thailand
Wen-Tsai Sung, National Chin-Yi University of Technology, Taiwan
Wojciech Zabierowski, Technical University of Lodz, Poland
Su Wu-Chen, Kaohsiung Chang Gung Memorial Hospital, Taiwan
Yasin Kabalci, Nigde University, Turkey
Yoshiro Imai, Kagawa University, Japan
Zanila Omury, Dublin Institute of Technology, Ireland
Zuzing Zhu, University of Science and Technology of China, China

Overview

The SDIWC International Journal of Digital Information and Wireless Communications is a refereed online journal designed to address the networking community from both academia and industry, to discuss recent advances in the broad and quickly-evolving fields of computer and communication networks, technology futures, national policies and standards and to highlight key issues, identify trends, and develop visions for the digital information domain.

In the field of Wireless communications; the topics include: Antenna systems and design, Channel Modeling and Propagation, Coding for Wireless Systems, Multiuser and Multiple Access Schemes, Optical Wireless Communications, Resource Allocation over Wireless Networks, Security, Authentication and Cryptography for Wireless Networks, Signal Processing Techniques and Tools, Software and Cognitive Radio, Wireless Traffic and Routing Ad-hoc networks, and Wireless system architectures and applications. As one of the most important aims of this journal is to increase the usage and impact of knowledge as well as increasing the visibility and ease of use of scientific materials, IDIWC does NOT CHARGE authors for any publication fee for online publishing of their materials in the journal and does NOT CHARGE readers or their institutions for accessing to the published materials.

Publisher

The Society of Digital Information and Wireless Communications
Miramar Tower, 132 Nathan Road, Tsim Sha Tsui, Kowloon, Hong Kong

Further Information
Website: http://sdiwc.net/ijdiwc, Email: ijdiwc@sdiwc.net,
Tel.: (202)-657-4603 - Inside USA; 001(202)-657-4603 - Outside USA.

Permissions

International Journal of Digital Information and Wireless Communications (IJDIWC) is an open access journal which means that all content is freely available without charge to the user or his/her institution. Users are allowed to read, download, copy, distribute, print, search, or link to the full texts of the articles in this journal without asking prior permission from the publisher or the author. This is in accordance with the BOAI definition of open access.

Disclaimer

Statements of fact and opinion in the articles in the International Journal of Digital Information and Wireless Communications (IJDIWC) are those of the respective authors and contributors and not of the International Journal of Digital Information and Wireless Communications (IJDIWC) or The Society of Digital Information and Wireless Communications (SDIWC). Neither The Society of Digital Information and Wireless Communications nor International Journal of Digital Information and Wireless Communications (IJDIWC) make any representation, express or implied, in respect of the accuracy of the material in this journal and cannot accept any legal responsibility or liability as to the errors or omissions that may be made. The reader should make his/her own evaluation as to the appropriateness or otherwise of any experimental technique described.

Copyright © 2015 sdiwc.net, All Rights Reserved

The issue date is January 2015.
## TABLE OF CONTENTS

### Original Articles

<table>
<thead>
<tr>
<th>PAPER TITLE</th>
<th>AUTHORS</th>
<th>PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>AN IN-MEMORY APPROACH TO SENTIMENT ANALYSIS BASED ON SAP HANA</td>
<td>Karl Kurbel, Dawid Nowak, Florian Jatzold, Pavlo Glushko</td>
<td>1</td>
</tr>
<tr>
<td>MANUAL LOGIC CONTROLLER (MLC)</td>
<td>Claude Ziad Bayeh</td>
<td>21</td>
</tr>
<tr>
<td>SPATIAL SPECTRUM UTILIZATION EFFICIENCY METRIC FOR SPECTRUM SHARING SYSTEM</td>
<td>A. A. Ayeni, N. Faruk, N. T. Surajudeen-Bakinde, R. A. Okanlawon, Y. A. Adediran</td>
<td>44</td>
</tr>
<tr>
<td>CIPHERTEXT DIVERGE-MERGE SCHEME OF IDENTITY-BASED ENCRYPTION FOR CLOUD-BASED FILE TRANSMISSION SERVICE</td>
<td>Makoto Sato, Masami Mohri, Hiroshi Doi, Yoshiaki Shiraishi</td>
<td>52</td>
</tr>
</tbody>
</table>
An In-Memory Approach to Sentiment Analysis Based on SAP HANA

Karl Kurbel1, Dawid Nowak2, Florian Jätzold3, Pavlo Glushko4
European University Viadrina, Große Scharrnstraße 59, 15230 Frankfurt (Oder), Germany
1kurbel.bi@europa-uni.de, 2danowak@europa-uni.de, 3florian.jaetzold@gmx.de, 4pavloglushko@gmail.com

ABSTRACT

Sentiment analysis (SA) is one of a number of fields in which "big data" requires processing. The term "sentiment" refers to the feelings, beliefs, emotions, and opinions that people express in relation to certain matters in written form, more often than not in online media such as social networks. Sentiment analysis can provide enterprises with valuable insights into, for example, how customers regard their products. When sentiments are collected from social networks over a period of time, the amount of data can indeed be "big". This paper describes and evaluates a lexicon-based SA solution, implemented as a prototype using SAP HANA. Sentiments are collected from Facebook and Google+, using a combination of tools including Microsoft Excel and the Power Query add-on to Excel. As HANA employs in-memory database technology, the calculation of sentiment values will be very fast. As a result of the sentiment analysis, five categories (from strong negative to strong positive), plus a category "sentiment not found", are filled with individual sentiments and, with the help of business intelligence tools, evaluated as well. SA results are made available on a user interface created using SAP's current interface technology, SAPUI5.

KEYWORDS

Sentiment analysis, lexicon, in-memory database, SAP HANA, social networks

1 INTRODUCTION

Social media provide new tools for the efficient creation and sharing of ideas between people [1, p. 15]. For those people who wish to publicly share their opinion on the Internet, easy-to-use information and content sharing tools are available, such as microblogs, social networks, and discussion forums. With the help of these tools, social media users can exchange ideas and opinions in a time- and cost-effective manner. Taking into account the number of statements that are posted every day on social networks such as Twitter, Facebook, or Google+, the volume of data containing opinions, feelings, ideas, counter-statements, etc. is extremely large [2]. A term currently used to denote this volume is "big data" [3]. For both businesses and other organizations, it is possible to gain valuable insights from these posts, as they convey the subjective perceptions of individuals. For example, a discussion about a product recently released onto the market can enable the vendor to assess consumers’ opinions of the product, how the target group regards the product, what features consumers like or dislike, and ultimately help them improve the product. To best utilize information hidden in postings useful to the company's decision makers, this information must be extracted from unstructured streams of data and subsequently analyzed. Sentiment analysis (SA) is a method for examining unstructured social data. It is also known as opinion mining [2]. Sentiment analysis identifies and extracts people’s sentiments from the source data using natural language processing and other text analysis techniques [4]. ‘Sentiments’ is taken to refer to people’s moods and their feelings, beliefs, emotions, and reviews, as expressed in written form [5, p. 7]. The aim of a sentiment analysis is to acquire knowledge about polarities on a specified topic (e.g. a product, service, organization, individual, issue, etc.), using the information collected.
The sentiments are then assigned to a category, e.g. positive, negative, or neutral. Some tools are capable of sorting sentiments into more specific categories [6] [3]. Polarities are identified as groups of sentiments [7, p. 2]. Companies have invested in developing their own opinion mining solutions. These solutions have been used to analyze products and services in healthcare, finance and other areas and in some cases even to make predictions of political election results [5, p. 9] [8] [9, pp. 1170-1172].

A large data volume is a prerequisite for meaningful results. In the original form, this data is unstructured. The data has to be structured and stored in a database in order to be able to perform a sentiment analysis. The compiling of the database is a task that requires powerful tools to handle and store large volumes of unstructured data.

In-memory technology is an enabler for the processing of big data in general and for sentiment analysis in particular. The price reduction in the market for memory chips and the progress in multi-core processors have made possible the development of powerful in-memory database solutions. Their main advantage over disk-based databases is an extremely fast data access rate. Based on this, organizations are able to act and react to the market very quickly, as up-to-date information is available almost in real-time. To indicate this development, the term real-time enterprise has been coined. For sentiment analysis, in-memory databases help to improve both the quality of the results and performance.

This paper describes and evaluates the design and implementation of a sentiment analysis solution using an in-memory database. Sentiments were collated from social media, in particular from Facebook and Google+. The aim of this paper is to provide insights into the mechanisms of a sentiment analysis tool, based on SAP HANA, and to point out possible avenues of investigation for further improvement. SAP HANA is an "appliance", which means that it consists of both hardware (for hosting and processing in-memory databases) and software – including a DBMS (database management system), an IDE (integrated development environment), and application and web servers, among others [10].

In the next section, the state of the art in the field of sentiment analysis is briefly outlined. Section 3 goes on to describe the architecture of the prototypical SA solution, the process of sentiment acquisition from social networks, sentiment calculation, and the user interface. The fourth section provides results of performance tests carried out on SAP HANA and on a standardized SQL server. In section 5, the results are evaluated, then the final section provides a brief summary and an outlook for future research.

2 RELATED WORK

Sentiment analysis has become a popular field of research for various stakeholders [11]. Linguists, computer scientists and managers benefit from a steady progress in the field [5, p. 10] [12]. Due to the potential that sentiment analysis has for real-time business, research into the field is ongoing [9]. Research into technological factors such as in-memory technology and column-oriented data storage [13] helps to improve SA, by significantly reducing the time required for data querying [14].

Since there is an abundance of online contributions available on social networks, the essential challenge of sentiment analysis is to be able to filter out relevant contributions effectively. The application programming interfaces (APIs) of social networks (such as Facebook and Twitter) allow the user to sort through the mass of messages in order to identify those containing certain search terms. However, this is only the first step.

Messages can be defined as unstructured data. A major difficulty – and the core of an SA algorithm – is to extract sentiments from these data. Various approaches for this step have been proposed, ranging from static lexicon- and rule-based methods (preferred by industry) to more sophisticated solutions based on machine learning and natural language processing (propagated...
by academia) [15] [16] [17] [18, pp. 145, 191]. All approaches have their strengths and weaknesses, which can be viewed clearly in the quality of the analysis [7, p. 164]. A recent study demonstrated that the rule-based approach is more suited to practical questions than sophisticated machine learning or combined approaches [18].

3 SOLUTION ARCHITECTURE

The architecture of the sentiment analysis solution presented in this paper is outlined in figure 1. It consists of a cloud-based SAP HANA instance (HANA SPS 6 running on a multi-core HP server with 2 TB RAM and 64 cores), Microsoft Excel 2013 with the MS Power Query for Excel add-on (version 1.5), a SAPUI5 user interface, and Facebook and Google+ APIs.

Power Query is used to perform an ETL process (cf. section 3.1) on the social data collected from Facebook and Google+. These data are accessed through application programming interfaces provided by Facebook and Google+ and stored in the SAP HANA database using a column-oriented storage format.

The sentiment analysis solution was developed as an SAP HANA native application. HANA native applications are executed inside HANA. Since SAP HANA has a built-in application server (named SAP HANA Extended Application Services [19]), unlike a traditional database, the SA solution runs directly in HANA. The application server comes with a web server, which allows HTTP-based user interfaces to also be executed inside HANA. Since the application server and the database are integrated at a low level, communication between the database layer, the data-processing layer, and the presentation layer is simplified. This leads to better performance of the application than would be the case were an external application server used [19].

SAP HANA Studio, a dedicated development environment, was utilized to develop HANA-based artifacts and procedures.
3.1 ETL Process for Social Data

ETL (extract, transform, load) is necessary because unstructured data have to be read, interpreted, and transformed into structured data before they can be loaded onto the HANA database. This process has three steps: (1) Reading the raw data from the sources with the help of Facebook and Google+ APIs; (2) cleaning and formatting the data; (3) loading the data onto the database.

The input to the sentiment analysis consists of data on users' attitudes. In order to retrieve the data, APIs provided by the aforementioned social platforms were used. An API represents an understandable set of rules for the exchange of information between the platform and the developer's machine.

Working with an API follows the schema of a request-response process, in which particular formats for both the request and the response apply. A request, in the form of an HTTP GET, is sent from the developer’s device (PC, smartphone, online app, etc.) to the API's server via the Internet. The syntax of the request will be specific to each social network, though the general concepts remain the same. The response from the API arrives as a JSON file. JSON (JavaScript Object Notation) is a compact data format for data exchange between web services [20]. The file contains structured text, which requires parsing for information.

An example of a request using the Facebook API is as follows:

https://graph.facebook.com/search?q=windows8&type=post&fields=id,from.fields(id,name),message&access_token=24858H

where:

- https://graph.facebook.com/ is the URL of the Facebook API,
- search?q=windows8 indicates a search term, which should be contained in every record retrieved as a response (here given as "windows8"),
- fields=id,from.fields(id,name),message indicates those fields to be included in the response (here a unique id of the post, the user’s id and name, and the text message),
- access_token=24858H is the access key allowing the developer to retrieve information via the API.

APIs enable the developer to retrieve almost in real-time large volumes of social data. However, they are unable to effectively filter this data during the retrieval process, as logical operators (e.g. AND, OR, NOT) are not available. This leads to excessive amounts of unnecessary information being collected.

Another disadvantage of the Facebook and Google APIs is the scarcity of publicly available documentation. Thus, we have been forced to look into necessary capabilities by trial and error or by sourcing information from web forums.

The term "public posts", which refers to those posts examined in sentiment analysis, is a problematic one, given the varying ways in which the term may be interpreted. What it stands for will depend on the social network in question. For the purpose of this paper, a public post is defined as any content posted by the user that is accessible to all other users registered on the same social network, without requiring any additional access rights. As a post can be a text, a song, a video, a picture, or any other multimedia format, the content of the posts must be filtered in order that only posts containing textual information are obtained.

Our first design of the ETL process was based on HTTP requests and a JavaScript application provided inside the SAP HANA XS application engine. Had the JavaScript application functioned as documented, it would have loaded a JSON file as a response from the API with the help of an HTTP remote request; parsed, cleaned, and formatted the data; and finally...
loaded it onto a database table. However, due to reasons that are still not entirely clear, the outbound HTTP request did not work for our software and hardware setting. Since the JSON file could not be read directly by the JavaScript application, a potential solution could have been to create a JSON file externally (i.e. outside the HANA system). This file might then have been exported to the HANA server and parsed/loaded using the JavaScript application. However, this solution appeared overly complicated and inconvenient, as end users would have been forced to employ several different tools for the ETL process.

Since ETL automation within HANA was not possible, a different approach was therefore required. In fact, an add-on to MS Excel, Power Query, provides the features needed. Hence, the social data were collected with the help of Power Query, stored in an Excel spreadsheet, and then exported as a comma delimited text file (.csv) to HANA. In HANA, the .csv file was imported into a database table (named SocialData table).

The SocialData table serves as the input to the sentiment calculation algorithm (cf. section 3.2). It contains columns for the ID of a social message, the ID of the user who posted it, the

---

**Figure 2.** ETL process

---

<table>
<thead>
<tr>
<th>MS Excel with VBA</th>
<th>Power Query</th>
<th>SAP HANA Studio</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>START</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Define new</td>
<td>Prepare and send queries based on the defined tasks</td>
<td></td>
</tr>
<tr>
<td>sentiment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>analysis task</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>[NO]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specify the</td>
<td>Queries sent to APIs</td>
<td></td>
</tr>
<tr>
<td>search term</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specify the</td>
<td></td>
<td></td>
</tr>
<tr>
<td>social channel</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transform the</td>
<td>Clean and format the social data</td>
<td></td>
</tr>
<tr>
<td>data into a CSV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>file</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>[YES]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>STOP</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend

- Activity
- Signals

**Figure 2.** ETL process
date of the post, the message text, as well as a sentiment value. Since the sentiment value is calculated later on in the process, it is initially NULL.

As shown in figure 2, the ETL process is as follows:
1. The user specifies the search terms and social networks to be queried by Power Query.
2. The user initiates the gathering of the data. Since the social networks' APIs do not provide substantial filtering and transformation features, Power Query functionality is used to clean and format the data.
3. Once the data have been prepared, they are transformed into a comma delimited file with the aid of a VBA script.
4. Finally, the user commits the .csv file to the server in the SAP HANA Studio IDE (integrated development environment). With the help of the IDE's load file service, the data in this file are automatically appended as records to the database table.

This process requires several manual actions from the user. On the other hand, a familiar software tool is employed (MS Excel), which means that the current solution can be taken up by Excel users with relative ease.

### 3.2 Sentiment Calculation Algorithm

Our solution uses a lexicon-based approach to sentiment calculation. This approach consists of a lexicon containing sentiment terms and values, which assign a term to a category. Sentiment terms are taken to be words that users may employ to express their feelings or opinions on a particular issue (e.g. "love", "fine", "error"). Five sentiment categories (strong negative, negative, neutral, positive, and strong positive) were defined. Sentiment values range from -2 to +2, with -2 standing for strong negative, -1 for negative, 0 for neutral, +1 for positive, and +2 for strong positive. In addition, a category for "Sentiment not found!" was introduced, which applies to all messages in which the algorithm could not identify a matching term from the lexicon. The message thus receives the value NULL.

To determine the overall sentiment in a message, the algorithm scans the message for sentiment terms. Based on the individual sentiment values, the algorithm calculates an average to represent the overall sentiment of the message. The lexicon-based approach was chosen because it is possible to adapt the process to the user's requirements with little difficulty. It can be quickly implemented, as opposed to machine learning technologies [15, p. 829], and is relatively simple to understand. Calculating averages makes it possible to assign an overall sentiment value to a message. This is not to deny that there are alternative methods for counting words from a lexicon in a message, and the impact of alternative methods on the algorithm's performance could be tested in future research. Using the category "Sentiment not found!" offers several advantages: First, it allows the quality of the results to be properly evaluated, as the share of unclassified messages is made explicit. If these messages constitute a substantial portion of the total number collected, the user will know that the results may not represent the actual distribution of sentiments. Second, attempts to extend the lexicon may be initiated from an examination of these messages, because it is reasonable to assume that they contain sentiment terms that are yet to be included in the lexicon. Third, if the number of messages in this category is small, the user is able to re-categorize them manually.

The list of sentiment terms was created based on a literature review [21] and the authors' own experience. Before the developer of the lexicon assigned a sentiment value to a term, he discussed the assignment with two other project participants.

All terms are written in lower-case letters. 20 percent of the terms represent positive or strong positive sentiments, 48 percent represent negative or strong negative sentiments, and 32 percent are neutral terms.

The algorithm calculating an average sentiment value for a message works as follows: First, it
determines the frequency with which a word from the lexicon is used in an individual message. This is done by comparing the original message length with the length of the message after the $i^{th}$ word from the lexicon has been replaced with a space. For example, the original message "I love Windows8" would then become "I Windows8", after the lexicon term "love" has been replaced. The difference in length is then divided by the length of the word minus 1 (cf. equation 1). 1 is then subtracted from the word length as one space will be where the sentiment word was before [22].

$$NO_i = \frac{ML_i^{before} - ML_i^{after}}{WL_i - 1}$$ (1)

where:

- $NO_i$ – number of occurrences of the $i^{th}$ word in the message,
- $ML_i^{before}$ – message length before replacement of the $i^{th}$ word,
- $ML_i^{after}$ – message length after replacement of the $i^{th}$ word,
- $WL_i$ – length of the $i^{th}$ word.

Afterwards, the algorithm multiplies the number of occurrences of each term with its respective sentiment value and sums up the products for all $N$ lexicon terms found. The sum is then divided by the total number of occurrences of lexicon terms in the message (cf. equation 2):

$$SAVG = \frac{\sum_{i=1}^{N} (NO_i \cdot SV_i)}{\sum_{i=1}^{N} NO_i}$$ (2)

where:

- $SAVG$ – sentiment average of the message,
- $NO_i$ – number of occurrences of the $i^{th}$ word in the message,
- $SV_i$ – sentiment value of the $i^{th}$ word.

To prevent division by zero, the algorithm returns a NULL value if no lexicon term was identified in the message. Finally, the average is compared with the intervals in table 1 to determine the sentiment category of the message.

A database table is used to represent the lexicon in HANA, while the calculation algorithm is implemented as an SQLScript procedure. SQLScript is a set of SQL extensions that allows for placing data-intensive logic in a HANA database (and not on the application server). Scripts can be executed in parallel on multiple CPUs [23, p. 8].

**Table 1.** Sentiment categories and their intervals

<table>
<thead>
<tr>
<th>Sentiment category</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strong positive</td>
<td>1.6</td>
<td>2.0</td>
</tr>
<tr>
<td>Positive</td>
<td>0.6</td>
<td>1.5</td>
</tr>
<tr>
<td>Neutral</td>
<td>-0.4</td>
<td>0.5</td>
</tr>
<tr>
<td>Negative</td>
<td>-1.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Strong negative</td>
<td>-2.0</td>
<td>1.5</td>
</tr>
<tr>
<td>Sentiment not found!</td>
<td>NULL</td>
<td>NULL</td>
</tr>
</tbody>
</table>

The procedure used to calculate the sentiments is triggered when the user presses a button on the user interface, which then invokes a server-side JavaScript (XSJS) service. Server-side JavaScript is utilized for exposing the consumption model in SAP HANA [19, p. 319]. The functioning of the algorithm is illustrated in figure 3. The user triggers the algorithm by clicking the start button on the user interface.

For the first step, the algorithm selects all messages that have a NULL value in the "Sentiment" column of the input table (SocialData table), using a cursor. In other words, it selects all messages for which a sentiment value has not yet been determined.

The next step converts all characters to lower case using an SQL function. This is required as the lexicon contains only lower-case letters.

Once these initial steps have been completed, the average sentiment value of the message can be calculated, based on equations 1 and 2. SQLScript functions are used to replace words in
Select all messages from the SocialData table where the sentiment value is NULL

For each row in dataset do
    Convert the message text to lower case
    For each term in lexicon do
        Calculate the number of occurrences of the term in the message text [cf. Eq. 1]
        Next
    Calculate the sentiment average for the message [cf. Eq. 2]
    Assign a sentiment category to the message according to the intervals defined in the sentiment category table [cf. Table 1]
    Store the sentiment category in the sentiment value field of the SocialData table
Next

Figure 3. Algorithm for sentiment calculation

3.3 Creating the User Interface

In the context of sentiment analysis, the main task of the user interface (UI) is proper visualization of analytical data regarding the sentiment distribution [12, p. 63]. The user interface was developed with the help of SAP HANA Studio as a native HANA application. As such, it runs directly inside HANA using the built-in application and web servers. The front-end is an HTML page with JavaScript elements, which are responsible for the core functionalities of the UI. The web page is composed of preconfigured UI elements that come with the SAP HANA UI libraries (called SAPUI5 [24]). Communication between the presentation layer and the database layer is provided through server-side JavaScript services. These services then generate, in response to URL-based queries (REST services), output in the form of a JSON file.

As shown in Figure 4, the user interface contains two diagrams, which depict the sentiment distribution for the selected search term. The pie chart illustrates sentiment shares in relation to the search term, including the special category "Sentiment not found!". In addition, the column chart shows the sentiment distribution for those messages where sentiments were found. In the lower section of the user interface, details of the messages containing sentiments about the search term are included, in particular the message's text, the calculated sentiment, the date, and the ID of the user who wrote the message.

The search term ("windows 8") was selected from the drop-down list in the top left hand corner of the main pane. The list is automatically
filled with distinct search terms from a database table. Selecting a search term results in a query being sent to the service. The query includes the search term as one of the input criteria. The service then proceeds to query the database (using SQL). It returns a list with the sentiment categories and the sentiment counts (in the form of a JSON file). This list is used as input for creating or updating the two diagrams.

The "Refresh data" button on the command bar is linked to the sentiment calculation procedure described in section 3.1. Pressing the button initiates the procedure, which calculates sentiment values for new messages stored in the database. The other two buttons are used to update the calculated sentiments manually (such as altering the sentiment category) and to download the message details shown in the lower part of the screenshot as an Excel file (.xls).

The user interface presented in figure 4 is a rudimentary one, showing only the most important analytical data. However, its functionality can be easily adapted to user requirements with the help of pre-defined interface elements of the SAPUI5 libraries.

4 PERFORMANCE TESTING

In order to measure the performance of the sentiment calculation algorithm based on SAP HANA, several tests were carried out. For test purposes, 1,220 social messages making reference to the term “Windows8” were collected with the help of Facebook and Google+ APIs. After the ETL process had been performed, the messages were placed into the table, which constituted an input for the sentiment calculation algorithm. Next, using SAP HANA Studio, the sentiment calculation procedure was called and its execution time was measured. The execution time consists of data access time (time required to access the data in the database table), calculation time (i.e. algorithm) and data update time (time needed...
to save the calculation results back to the database table.

The tests were repeated several times and the average algorithm performance was then calculated.

It took approximately three seconds to access the sentiment data, then calculate and update the sentiment fields in 1,220 social messages (each up to 5,000 characters long), using a lexicon of 5,000 terms.

To better understand the test results, a comparison with a standard (non in-memory) database server was carried out. The same sentiment calculation algorithm was implemented on Microsoft SQL Server 2008 using the scripting language transact-SQL (T-SQL). For test purposes, a Microsoft SQL Server installed on a PC with 4 CPUs (2 GHz) and 4 GB of RAM was used.

Then, using a client application (Microsoft SQL Server Management Studio), the execution time of the stored procedure was measured. To make the measurements comparable, the same input data (as in the HANA case) was used.

For a more detailed analysis, the social messages that had been collated were divided into three groups sorted by message length:
- short messages, containing up to 150 characters,
- medium length messages between 151 and 1,000 characters, and
- longer messages containing more than 1,000 signs.

For test purposes, a representative sample of 100 messages from each group was chosen. The test used a constant lexicon containing 5,000 words.

Once this was completed, a comparative analysis of the algorithm’s execution time on SAP HANA and Microsoft SQL Server was performed for each of the above groups. Test results measured in seconds are presented in table 2.

The tests showed that the calculation is approximately 10 times faster when based on SAP HANA than on a conventional database server.

### Table 2. Execution times of the sentiment calculation algorithm in relation to the message length

<table>
<thead>
<tr>
<th>Message length [No. of characters]</th>
<th>SAP HANA [sec]</th>
<th>SQL Server [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 150</td>
<td>2.2</td>
<td>19.8</td>
</tr>
<tr>
<td>&gt; 150 and &lt;=1,000</td>
<td>3.9</td>
<td>39.4</td>
</tr>
<tr>
<td>&gt; 1,000</td>
<td>15.6</td>
<td>178.1</td>
</tr>
</tbody>
</table>

The difference in performance becomes particularly clear in the case of longer messages. The sentiment calculation on HANA takes approximately four seconds, whereas the SQL Server requires nearly three minutes for calculating the results. In the case of short messages, the difference between the analyzed implementations is not significant (10 seconds), however it may become significant with a higher number of messages. Similar tests were also carried out to identify the relationship between the algorithm’s performance and the lexicon size.

In this scenario the execution times of the stored procedures implemented on SAP HANA and Microsoft SQL Server, respectively, was measured taking into account a variable size of the lexicon.

As input data for the test, all collected social messages (1,220) of different length were used. Performance of the sentiment calculation algorithm based on SAP HANA and Microsoft SQL Server in relation to the lexicon size is presented in the table 3.

### Table 3. Execution times of the sentiment calculation algorithm in relation to the lexicon size

<table>
<thead>
<tr>
<th>Lexicon size [No. of terms]</th>
<th>SAP HANA [sec]</th>
<th>SQL Server [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.8</td>
<td>3.3</td>
</tr>
<tr>
<td>50</td>
<td>1.1</td>
<td>7.4</td>
</tr>
<tr>
<td>100</td>
<td>2.9</td>
<td>13.2</td>
</tr>
<tr>
<td>500</td>
<td>5.8</td>
<td>53.2</td>
</tr>
<tr>
<td>1000</td>
<td>10.9</td>
<td>103.5</td>
</tr>
<tr>
<td>5000</td>
<td>48.0</td>
<td>512.4</td>
</tr>
</tbody>
</table>
The test results show that the lexicon size, as well as the volume of input data, has a direct impact on the algorithm’s execution time. Similar to the previous analysis, these tests also confirmed that the procedure’s performance on HANA is much better than on Microsoft SQL Server. When using a small lexicon (100 words) the difference in the algorithm’s execution time is approximately 10 seconds. With an increase in the lexicon size, the advantage of SAP HANA over Microsoft SQL Server also becomes greater. Using a lexicon containing 5,000 words, the difference in the algorithm run-time increased to 460 seconds.

The accuracy of lexicon-based sentiment analysis grows with increasing number of lexicon terms. In-memory database technology allows to use very large lexica, enabling almost real-time sentiment analysis.

Both tests clearly proved that sentiment analysis can be enhanced using in-memory database technology. Even better performance in comparison to the traditional databases can be expected with higher volumes of input data and larger lexicon sizes.

5 EVALUATION

The SA prototype described in section 3 has strengths as well as areas for improvement. With regard to the strengths, three characteristics are worth mentioning.

In contrast to solutions proposed by other authors, a category "Sentiment not found!" was introduced for messages containing none of the lexicon terms. While other solutions treat this situation as a neutral sentiment, we disregard messages in which none of the terms appear. In this way, messages without a sentiment have no impact on the results, thus improving their quality.

In addition, the HTML-based user interface provides easily comprehensible information for the end user. The interface has a dashboard format, which requires no special preparation or training from the user to obtain the information. Likewise, most business users are familiar with MS Excel, which they are compelled to use to define the search terms and the social media to be searched.

The sentiment analysis solution has benefited significantly from being developed as a HANA native application. Execution times required for the calculation, aggregation, and visualization of sentiments are very short in comparison to other solutions [14].

In order to measure the performance, several tests were carried out. The performance depended directly on the number and the lengths of the social messages and on the lexicon size. Good performance may also be expected with higher volumes of data because the solution is based on SAP HANA’s in-memory technology, using only one column-oriented database table for both the input and output of the results of the calculation.

However, the SA prototype outlined in this paper also showed some drawbacks requiring improvement in future versions. In its current form, which is based on SAP HANA (SPS 6), a significant amount of manual work is required for the ETL process and to initiate the sentiment calculation. The next version of HANA (SPS 7) provides features for automating ETL functionality inside the HANA system, without the need to resort to external tools [19, p. 464]. Instead of VBA code, SQLScript and Calculation Engine functions can then be used. With the aid of the new features, the application should be more convenient for the user and faster to work with.

Another area requiring improvement is the lexicon definition process. The current version enables the user to open and edit the lexicon exclusively with SAP HANA Studio. If it were possible to edit the lexicon (sentiment words and values) using a web front-end, the application would be more user friendly.

The most fundamental issue, however, is the sentiment calculation algorithm. Solutions for sentiment analysis are only as effective as their algorithms. While the computed averages
appear to be quite reliable, the classification of the individual messages into sentiment categories is not often reliable, i.e., messages may be put into a wrong category. However, this issue involves questions of natural language processing and linguistics, which are beyond the scope of this paper.

6 SUMMARY AND OUTLOOK

This paper outlined the development of a sentiment analysis solution with basic functionality, using SAP HANA in-memory database technology. The solution consists of three major sections: the ETL process, the algorithm for sentiment calculation, and the user interface.

Data from Facebook and Google+ were collated via the respective APIs, formatted, and then loaded into the database using an ETL process. The sentiment calculation algorithm determines the messages' sentiments using a lexicon-based approach. The lexicon was implemented as a database table in SAP HANA. The user interface used is an HTML page with JavaScript elements, which uses preconfigured SAPUI5 elements delivered with SAP HANA.

While the sentiment analysis on HANA allows for almost real-time analysis of social data, a drawback of the current solution is the manual ETL process, which requires user actions in Excel and Power Query. This issue could be resolved with the help of later versions of SAP HANA.

Further areas for improvement are threefold: First, the current solution only provides an elementary user interface. This UI could be extended with features for more effective data analysis. Second, the algorithm applies a very basic approach for sentiment detection by searching for occurrences of lexicon words—a process that could be improved at a later date by taking semantic structures into account. Third, the lexicon requires extending in order to increase the accuracy of the results and to broaden the application scope. In this context, word lengthening may also be considered [25].

In future research, the solution should be specifically tuned for SAP HANA. The present solution is not explicitly optimized for parallel execution, which is one of the strengths of SAP’s database technology. In addition, more sophisticated filtering techniques during data retrieval via the APIs would improve the solution's performance. For example, as logical operators are currently not supported, considerable amounts of unnecessary information are retrieved. Improving this situation, however, lies in the hands of the software vendor. In addition, natural language text recognition and processing remains a broad field for research, which could lead to enhanced SA solutions.

One limitation of this research is that it focuses exclusively on individual messages posted by social network users, without analyzing social internetworking scenarios [26]. An analysis of links between posts in the underlying networks, and of user roles and user impact on other users could provide additional information for business organizations [26, p. 202]. This may be considered another area for future research and extension of the existing SA solution.

ACKNOWLEDGEMENTS

The authors would like to thank Niraj Singh and Frank Finkbohner from SAP AG for their support and assistance with this project.

7 REFERENCES


Implementation of Resource Discovery Mechanisms on PeerSim: Enabling up to One Million Nodes P2P Simulation

Azrul Amri Jamal, Wan Suryani Wan Awang, Mohd Fadzil Abdul Kadir, Azwa Abdul Aziz
Faculty of Informatics and Computing,
Universiti Sultan Zainal Abidin, Tembila Campus, 22200 Besut, Terengganu, Malaysia.
(azrulamri, suryani, fadzil, azwaaziz)@unisza.edu.my

William John Teahan
School of Computer Science,
Bangor University, Dean Street, Bangor, Gwynedd, LL57 1UT, United Kingdom.
w.j.teahan@bangor.ac.uk

ABSTRACT

The objective of this study was to implement resource discovery mechanisms onto a Peer-to-Peer (P2P) simulator. Resource discovery mechanisms such as random walk, restricted random walk, breadth-first search, intelligent-BFS, depth-first search, adaptive probabilistic search, and blackboard resource discovery mechanism had been adopted onto PeerSim. The proposed implementation has enabled the simulation for resource discovery in P2P simulator which capable of running up to one million nodes. Brief information about each resource discovery techniques that are implemented onto PeerSim are discussed and presented.

KEYWORDS

Resource discovery, Peer-to-peer, Simulation, BRDM, PeerSim.

1 INTRODUCTION

Resource discovery is one of the most important parts in any resource sharing systems [1, 2]. Nowadays, the most widely used resource sharing methods are the peer-to-peer (P2P) systems. Resources shared in P2P systems were usually spread all over the Internet. Resource discovery plays a vital role in P2P, because it does not possess a central computer storing of all the resources.

A number of researches simulated the resource discovery techniques on a P2P network [1, 2]. Lazaro et. al. [1] focuses on resource discovery mechanisms that are used to find computational resources. The paper addresses the main requirements in decentralised resource discovery system such as, search flexibility, scalability, churn and fault tolerance, completeness, accuracy, security, and miscellaneous performance requirements.

Navimipour and Milani [2] analyse and examine resource discovery techniques into four main categorise such as, unstructured, structured, super-peer, and hybrid. The main requirements being examined were scalability, dynamicity, reliability, load balancing, response time, and robustness of the resource discovery techniques.

Lazaro et.al. [1], and Navimipour and Milani [2] did a comprehensive study on resource discovery techniques, however did not specify the simulators that being used in their researches. The paper also did not mention the amount of nodes being simulated for the researches.

The objective of this paper is to add several available resource discovery mechanisms onto PeerSim [3] so that it will provide a platform for future researches to do further simulation and testing of each developed protocols. Developing the mechanisms on PeerSim enables a vast amount of P2P nodes to be generated. Thus making the resource discovery mechanisms simulation nearer to the real-life application.
This paper is a continuation of a proceedings paper in The Third International Conferences on Informatics and Applications (ICIA2014) [4].

2 RESOURCE DISCOVERY MECHANISMS

Resource discovery techniques have been studied even before the existence of computer networks. The various techniques that being used today are the improvements of previously developed resource discovery methods. Techniques such as Random Walk [5], Restricted Random Walk [6], Breadth-First Search [7], Intelligent BFS [8], Depth-First Search [9], Adaptive Probabilistic Search [10], Blackboard Resource Discovery Mechanism [11] are explained briefly in the following subsections.

2.1 Random Walk (RW)

```
0  Receive search message;
1  if(this node has the resource){
2      Reply to originator;
3  }
4  Forwards the message to a random adjacent node;
```

FIGURE 2. RW Pseudocode

Random walk [5] is an easy method to locate resources. When one node is searching for a resource, the node will check for the resource at its current location. If the resource is not available, the node will send a walker (query) to one adjacent node. The selection of adjacent node the walker should go next is decided randomly, thus the name of the method.

The search will be done recursively until it finds the resource that it was looking for. There are no restrictions in RW, therefore there is a possibility that the walker will go back to the node that the walker has been visited previously [5]. The pseudocode for random walk is shown in Figure 2.

2.2 Restricted Random Walk (RRW)

Restricted random walk [6] represents an improvement upon random walk resource discovery mechanism. It carries most of the criteria of a RW, however, the only differences are when the walker (query) is selecting the adjacent node to go to.

RRW’s walkers will randomly select an adjacent node that it has never visited before. Therefore in order to run RRW, the walker will keep track all of the nodes that it has visited.
The ability to omit the nodes that it has visited makes the RRW a better method than RW because it does not waste the time-to-live (TTL) of its walkers. Figure 3 shows the pseudocode for restricted random walk.

### 2.3 Breadth-First Search (BFS)

BFS is among the earliest resource discovery techniques that being used in the field of computer networks. In P2P networks, when a node is searching for a resource, it will check itself whether it has the requested resource. If not, the node will query all adjacent nodes for the resources [7].

BFS uses a lot of networking resources by sending a large amount of queries inside the network. This characteristic makes BFS looks like it is flooding the network with queries. That is the reason why BFS discovery method is also known as flooding [6]. Figure 1a shows the sequence of walkers being generated using the BFS technique. Pseudocode for breadth-first search is shown in Figure 4.

### 2.4 Intelligent BFS (Int-BFS)

Intelligent BFS [8] is an advancement of BFS [7] searching technique. Int-BFS does not flood the entire network. Instead, it only sends walkers to a fraction of its adjacent nodes. The fraction changes according to the topography and the number of adjacent nodes.

In Int-BFS, the node will store query information regarding how many times that the adjacent neighbour has been answering the majority of queries sent by the node.

When a new query arrives, the node will search the stored queries that it has, and forwards it to a set number of neighbours that have answered the most results for the query. Figure 1c shows the sequence of walkers using Int-BFS. Observe that the technique uses random, therefore the results would not be like that in every simulation. Figure 5 shows the pseudocode for intelligent BFS.

### 2.5 Depth-First Search (DFS)

Depth-first search [9] can be viewed as the opposite of BFS [7]. Instead of forwarding queries to all adjacent nodes, DFS only forwards one walker to one adjacent node.
then continues to go forward to another adjacent node. 

Figure 1b shows the route sequence that a DFS walker would choose during the resource discovery. Observe that the technique uses random therefore the results might differ every simulation cycle. Figure 6 shows the pseudocode for depth-first search.

2.6 Adaptive Probabilistic Search (APS)

Adaptive probabilistic search [10] is a modification of random walk [5]. In the initial stage, APS works just like RW, where all adjacent nodes have an equal probability to be selected. The probability of an adjacent node increases when it returns a hit for any query. Alternatively, the probability to be selected decreases if the adjacent node did not return any successful searches [10]. For this reason, APS searching capabilities improve over time.

Pseudocode for adaptive probabilistic search is shown in Figure 7.

2.7 Blackboard Resource Discovery Mechanism (BRDM)

Blackboard resource discovery mechanism was first coined in 2004 as a method that utilises an  

Figure 8. BRDM Pseudocode

The selection of the forwarding node is done randomly. The query will continue on forward until it could not find any other adjacent node that it has not visited. When it reaches the end (where there is no other choice), the walker will take one step back to the previous node that it visited, and
artificial intelligence technique called blackboard [11]. The blackboard technique is when we list down all the important information about neighbouring entity.
In BRDM, the blackboard is used to list down recommendable adjacent nodes to forward a future query. If there is a recommended node, the query will be forwarded to the node, if there aren’t any recommended nodes, the query will be forwarded to a number of random adjacent nodes.
BRDM forwards queries using walkers. The amount of walkers is decided based on the TTL of the query and the amount of adjacent nodes. The percentage of neighbour to forward the query can be modified to suit the topology of the P2P network.

BRDM is utilised as one of the efficient scheduling policy for ParCop, a decentralised P2P system [12]. Figure 1c shows the sequence of walkers using BRDM. Observe that the technique uses random, therefore the results would not be like that in every simulation. Figure 8 shows the pseudocode for blackboard resource discovery mechanism.

3 PEERSIM & HARDWARE REQUIREMENTS

PeerSim [3] is a Java-based P2P network simulator. The simulator excels in generating large number of nodes (up to 1 million) in order to test and simulate a real-life P2P network. PeerSim project was initiated under EU projects BISON and DELIS. PeerSim development is now partially supported by Napa-Wine project. PeerSim has been developed with extreme scalability and dynamicity in mind. It consists of two simulation engines, one is cycle-driven and the other one is event-driven. In order to choose the right engine for the simulation, one needs to figure out which characteristics are the most important in the simulations that are about to be done.

3.1 Cycle-Driven Engine

Cycle-driven engine is simplified assumptions of a P2P network. It ignores the details in the transport layer of the OSI network model.

The reason for omitting the transport layer simulation is to enable scalability of the network, and to use as low computing resources as possible for the simulation. Cycle-driven engine is less realistic when compared to the event-driven simulation engine.

3.2 Event-Driven Engine

The event-driven engine was developed in order to cater to a more realistic simulator. Unlike the cycle-driven engine, the event-driven engine also enables OSI transport layer simulations.
In order to provide more realistic simulations than the cycle-driven engine, the event-driven engine is less efficient and uses a lot of computing resources. The amount of nodes can be simulated in this engine is usually lesser than cycle-driven engine, and the simulations are generally slower. All cycle-driven simulations can be implemented in the event-driven engine, but not vice versa.

3.3 Reasons For Choosing PeerSim

The main reason for choosing PeerSim as the simulator platform is because PeerSim is able to simulate large number of P2P nodes. This ability is important in P2P network simulation because it will truly reflect the real life environment of a vast P2P network [13].
The other reason for choosing PeerSim is because it is supported by several international fundings and programming community. The software has been used in P2P researches, and the ability to utilise either the cycle-driven or event-driven engine for its simulation is very useful.

3.4 Hardware Requirements

We have successfully run the one million nodes simulation on a computer with 4-core processors with two threads on each processor, and 16GB of random access memory. A Java Virtual Machine (JVM) is also needed to run PeerSim.
We have increased the JVM maximum memory heap size up to 10GB in order to run the simulation up to one million nodes.
4 EXPECTED RESULTS

We expect that the results will be coherent with the claims that each resource discovery technique claims. We will be able to get the data on the network overhead, successful and unsuccessful searches, average TTL of each resource discovery techniques. Table 1 shows the network overhead and successful searches of each resource discovery mechanism in a random P2P topology.

By utilising PeerSim as the platform, we can upscale the simulation to be running several thousand nodes, compared to other simulators that usually run to the maximum of a couple of hundred nodes.

Table 1. Network overhead and successful searches of each resource discovery techniques in a random P2P topology.

<table>
<thead>
<tr>
<th>Techniques</th>
<th>Network Overhead</th>
<th>Successful Searches</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Walk (RW)</td>
<td>★★☆☆☆</td>
<td>★</td>
</tr>
<tr>
<td>Restricted Random Walk (RRW)</td>
<td>★★★★☆</td>
<td>★☆☆☆☆</td>
</tr>
<tr>
<td>Breadth-First Search (BFS)</td>
<td>★★★★☆☆</td>
<td>★★★☆☆</td>
</tr>
<tr>
<td>Intelligent BFS (Int-BFS)</td>
<td>★★★★☆</td>
<td>★★★☆★</td>
</tr>
<tr>
<td>Depth-First Search (DFS)</td>
<td>★★★★</td>
<td>★★★☆☆</td>
</tr>
<tr>
<td>Adaptive Probabilistic Search (APS)</td>
<td>★★★★☆</td>
<td>★★★☆☆</td>
</tr>
<tr>
<td>Blackboard Resource Discovery Mechanism (BRDM)</td>
<td>★★★★★</td>
<td>★★★☆☆</td>
</tr>
</tbody>
</table>

5 CONCLUSION

The number of nodes in a real life P2P environment can be up to tens of million nodes. The number of P2P users keeps on increasing as the number of Internet users increase. Previous simulations usually were done with hundreds of nodes. Those simulations might come out with a different result when being adopted into the real life P2P environment. This is because some problems, such as network flooding might not be easily detected in a small P2P simulation compared to a large P2P simulation. By adopting resource discovery mechanisms onto PeerSim, the simulation can be run up to one million nodes. This capability enables researchers to test their resource discovery techniques to as near quantity as possible to the quantity in a real life P2P network environment.

The implementation of resource discovery techniques onto PeerSim will be the first landmark in making a low-cost, and higher amount of successful searches resource discovery mechanisms.

6 ACKNOWLEDGEMENTS

Authors, Azrul Amri Jamal and Wan Suryani Wan Awang would like to express their gratitude to Malaysian Ministry of Education (MoE) and Universiti Sultan Zainal Abidin (UniSZA) for sponsoring their PhD studies in Bangor University and Cardiff University, respectively. The authors would also like to thank UniSZA for preparing an environment conducive to research and development.

7 REFERENCES


Manual Logic Controller (MLC)

Claude Ziad Bayeh
EEE GROUP-R&D department
LEBANON
Email: c.bayeh@eee-group.net
claude_bayeh_cbegrdi@hotmail.com

ABSTRACT

The “Manual Logic Controller” also called MLC, is an electronic circuit invented and designed by the author in 2008, in order to replace the well known PLC (Programmable Logic Controller) in many applications for its advantages and its low cost of fabrication. The function of the MLC is somewhat similar to the well known PLC, but instead of doing it by inserting a written program into the PLC using a computer or specific software inside the PLC, it will be manually programmed in a manner to have the same functionality of the PLC. The program is done manually by connecting wires between coils and contacts of relays, in a way to form a program without using any memory or microcontrollers. It gives the same functionality of the PLC for small and medium programs, and moreover the contacts can support high currents, not like the PLC which supports only small current rate on its contacts. The lifetime of the MLC is much longer than the PLC, and its maintenance is much simpler, because anyone can make it or repair it. The MLC will not be defected by the variation of the input voltage and the input frequency because it is fabricated using mechanical relays which perform even if the electricity is not stable. The Grafcet and the Ladder parts of the MLC, are much simpler than the PLC for some applications, therefore, MLC is highly recommended for small and medium applications for its advantages over the PLC which are presented in this paper.

KEYWORDS

Manual program, PLC, Logic, Relays, Programming, Grafcet.

1 INTRODUCTION

In Electrical Engineering, a Programmable Logic Controller, PLC or Programmable Controller is a digital computer used for automation of electromechanical processes. It is used to control electrical devices and electromechanical devices such as machines, Air Conditioning systems, Lifts, electronic circuits, motors, Cars, etc. and any electrical or electronic devices [1-16]. The PLC has many inputs and many outputs, the number of inputs and outputs depend on the chosen size. The PLC is programmed using specific software. Every manufacturer has specific software for its PLC, the software varies from one manufacturer to another, but the concept is the same. The programming part of the PLC is done using computer and an interface that download the program from the computer to the PLC. The PLC has many advantages, but it also has many disadvantages which are presented in this paper.

In this paper, the author invented and designed a new circuit called MLC (Manual Logic Controller); the MLC can be programmed manually by connecting wires to coils and contacts of internal relays in a way to form logic operations (such as AND, OR, XOR, XNOR … and many others), which are equivalent to a written program on a PLC. The formed logic operations in MLC lead to a perfect functioning of the inputs and outputs similar to the PLC, but the difference is that the PLC can’t support high currents (for example 2A, 5A, etc…), whereas the MLC can support high currents, because it is made by using mechanical contacts with high rate of current capacity. There are many other advantages for the MLC which are presented in this paper. For instance, we can say that the MLC can replace the PLC for small and medium applications in which a simple program is sufficient to do the task. The price of the MLC is much cheaper compared to a similar PLC. The lifetime of a MLC is much
The maintenance of the MLC is much easier than the PLC. The cost of maintenance of the MLC is much lower than the PLC, and so on. This paper presents a comparative table between the MLC and PLC in order to show the advantage and disadvantage of each technology.

2 DEFINITION OF THE MANUAL LOGIC CONTROLLER (MLC)

The Manual Logic Controller (MLC) is a circuit invented and designed by the author in 2008 in order to replace the existing PLC for many applications when the need of a few logic operators is mandatory, and when a high current capacity on the inputs and outputs is needed.

The PCB circuit of the MLC is presented in figure 1:

The Manual Logic Controller (MLC) is a circuit invented and designed by the author in 2008 in order to replace the existing PLC for many applications when the need of a few logic operators is mandatory, and when a high current capacity on the inputs and outputs is needed.

The PCB circuit of the MLC is presented in figure 1:

Where:

- “I” indicates the input, for example I1 is the first input.
- “COM” indicates the common.
- “NO” indicates the normally open contact of the output relays.
- “NC” indicates the normally closed contact of the output relays.
For example, in the figure 1 the MLC has 29 inputs (I1, ..., I29), 9 outputs with 9 Normally open circuits (NO1, ..., NO9) and 9 Normally Close circuits (NC1, ..., NC9), and 7 intermediate Relays (RI1, RI2, ..., RI7).

Remark: The PLC has only one normally open contact for each relay, while the MLC can have many normally open contacts and normally closed contacts for each relay. Figure 1 presents only one NO contact and one NC contact for each output relay, but one can design an MLC with output relays that contain more than one NO contact and NC contact.

- Lines in blue color represent the internal connections between different relays, inputs and outputs. They are built in, which means one can’t change their connections.
- The small circles next to each other (vertically or horizontally) represent a pin holder or bus connector, it is used to connect flexible wires in order to do manual programming.
- For each input and each output, there is a wire connector (Figure 2); it is used to connect wires in the inputs and outputs.

2.1. Components of the MLC

There are three main components of the MLC, the first one is the Input, the second one is the intermediate relays which are used to program manually, and the third one is the Outputs.

Input:
Each input is connected to a bus connector with 5 possible connections or more. It is depicted in figure 2.

These inputs are used similar to the PLC inputs, but the difference is that they are able to support high current. The bus connectors are used to connect wires with other inputs, intermediate relays, and outputs in order to program manually the desired task.

Intermediate relays and connectors:
The intermediate relays (figure 3) are used to program manually the desired task by using wires connected between input bus connectors, intermediate relays and coils of the output relays. The form of connections may vary from one program to another depending on the desired task.
In figure 3, an extracted picture of the intermediate relays and bus connectors is presented. Each contact in the relays is connected to a small pin holder in the bus connector. For the presented case, each intermediate relay has 2 NO contacts and 2 NC contacts, it is possible to design a MLC with more NC and NO contacts. The intermediate bus connectors can be used as extra connectors that may help the programmer to connect more connections for each contact of the intermediate relays.

It is important to note that in a bus connector of an input and intermediate bus connector, all pin holders are connected together, which means that if one pin holder contains voltage, this means that all other pin holders in the same bus connector have the same voltage.

**Example of connection using intermediate relays:**

![Diagram](Image)

**Figure 4.** Example of programming manually a task on the MLC. The used wires for programming are in green color.

In figure 4, on the left side, wires in green color present the input of the MLC similar to the input of the PLC. In the middle, wires in green color are connected between the bus connectors of the inputs, the intermediate relays, the intermediate bus connectors and bus connectors of the outputs in a way to program manually a certain task, the same program can be written in a PLC using a computer. In the right side, wires represent the output of the MLC in which it controls external electrical or electronic circuits to work as the programmer wish.

**Output:**

The output relays are connected to external circuits through connectors (figure 5). Each contact of a relay is connected to a separate connector, in this paper the used relays have only 1 contact NO (normally open) and one contact NC (normally close). The current flows according to the status of the Relay, if the relay is in the normally close status, then the current flows from COM to the NC contact and output (case of relay R1 in figure 5). If the relay is in the normally open status, then the current flows from COM to the NO contact and output (case of relay R2 in figure 5).

![Diagram](Image)

**Figure 5.** Output Relays and current flow.

**2.2. How does the MLC work?**

1- The MLC receives signals (On/Off) from the inputs.
2- The form of connected wires between inputs, intermediate relays, intermediate connectors and coils of the output relays determine how the MLC works.
3- Intermediate relays will take status according to the input signals and each of them will have NC or NO contacts according to the designed form of the program.
4- The intermediate relays will order the output relays to take status such as NO or NC.
5- When the output relay takes certain status (NO or NC), it will let the current flows from the COM contact to the desired circuit connected to NO or to NC of the relay.

6- When the current flows from COM to a contact such as NC or NO, the connected equipment will work or will stop according to its connection.

### 2.3. Difference between PLC and MLC

In this sub-section, a comparative table is presented in order to compare the common points and different points between the MLC and the PLC.

<table>
<thead>
<tr>
<th>Description</th>
<th>PLC</th>
<th>MLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>How is it programmed?</td>
<td>Using computer</td>
<td>Manually using wires</td>
</tr>
<tr>
<td>Simplicity of programming</td>
<td>easy</td>
<td>easy</td>
</tr>
<tr>
<td>Can be expandable</td>
<td>Yes but the cost is high</td>
<td>Yes but the cost is low</td>
</tr>
<tr>
<td>Normally Open Contacts</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Normally Close Contacts</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>Write many programs in the same time</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>Current on the inputs and outputs</td>
<td>In Milli Amperes</td>
<td>From milli Amperes to Amperes</td>
</tr>
<tr>
<td>The hardware can be made by any person</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>Price of the device</td>
<td>&gt;$150 up to several thousands of dollars</td>
<td>$10 -$40</td>
</tr>
<tr>
<td>Armored for severe conditions (dust, heat…)</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Ratio (Price/Number of inputs)</td>
<td>Very high</td>
<td>Very low</td>
</tr>
<tr>
<td>Ratio (Price/Number of outputs)</td>
<td>Very high</td>
<td>Very low</td>
</tr>
<tr>
<td>Current rate used on the output contacts</td>
<td>Very small</td>
<td>Large</td>
</tr>
<tr>
<td>Use of extra mechanical relays to support high currents on the output</td>
<td>YES</td>
<td>Maybe</td>
</tr>
<tr>
<td>Analog inputs</td>
<td>YES</td>
<td>No, but it can be included</td>
</tr>
<tr>
<td>Digital inputs</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Contain Microprocessor</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>Contain complicated electronic circuits</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>reading speed of the inputs</td>
<td>In milli seconds</td>
<td>Instantly</td>
</tr>
<tr>
<td>Used for simple and medium complexity programs</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Used for very complicated programs (including analog sensors and complicated timers)</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>Complexity of fabrications</td>
<td>Very Complex</td>
<td>Very simple</td>
</tr>
<tr>
<td>Damaging probability</td>
<td>High</td>
<td>Low</td>
</tr>
</tbody>
</table>

### Table 1, Comparison between the MLC and the PLC.

#### 2.4. Advantages of the MLC over the PLC

As described in the table 1, the advantages of the MLC over the PLC are presented in bold red color. One can see clearly that the MLC has more than 17 advantages over the well known PLC.

#### 2.5. Advantages of the PLC over the MLC

As described in the table 1, the advantages of the PLC over the MLC are presented in bold green color. One can see clearly that the PLC has very few advantages over the MLC. But these advantages are important for very complicated programs.

### 3 RESULTS AND APPLICATIONS

In this section, the author is going to apply the same program and task using the MLC and the PLC, and analyzes the difference between programming using MLC and PLC. The following programming is a real case developed for KORMAK TS 33 machine, it is impossible to present all circuits and the complete diagrams for the machine, and it is somewhat difficult to understand how its circuits work, but in this paper, only little information is presented just to give simple idea about the machine. The goal of this section is to apply PLC and MLC programming to improve the functionality of the desired machine.

#### 3.1 Description of KORMAK TS 33 machine

The KORMAK TS 33 machine (figure 6-10), is an old machine that is used to strip and cut wires, which help manufacturers especially lighting manufacturers to speed up their work, when they need stripped wires within a specified length, the machine is able to produce around 120 stripped wires in one minute.
In this sub-section, it is impossible to explain in few lines how such complicated system works, but the goal of this subsection is to do programming using PLC and compare it with the programming using MLC. So many terms will not be explained.
Definition of the switches in figure 11:

(1) **Mains voltage** (B1): It indicates when the thermal switch of the machine is on (lamp indicator), thus enabling the connection of the electrical system.

(2) **Motor start switch MSS** (B3): To be pressed to operate the second motor which feeds cable to the main machine. The second motor will turn if it is pressed.

(3) **Automatic cycle button ACB** (B5): to be pressed in order to obtain a continuous production of processed cables.

(4) **Manual cycle button MCB** (B7): To be pressed for starting the cycle. It is used to verify the correct length of the cable and to verify the good functioning of mechanical and electrical parts; in general it is pressed before starting of an automatic cycle.

(5) **Operation button OB** (B2): To be pressed for operating the machine and enabling air supply. When it is pressed the position of pistons must returns to initial position.

(6) **Length reset button RB** (B4): To be pressed for transferring immediately the preset length into the electronic storage of the machine. If we finish a cycle that gives a length of 10 cm, and we want to begin a new cycle with a new length (for example 15 cm), we press the reset button to give the first cable a new length, if we don’t press reset button the first cable takes the initial length and all next cables take the new length.

(7) **Preset work piece light** (lamp indicator (B6)): It indicates to the operator that the machine has ended the preset work. When the preset work piece counter arrives to zero the light is illuminating to indicate that the cycle is finished.

(8) **Stop button** (EMERGENCY) **SB** (B8): To be pressed in case of emergency, when the need of stopping the machine is mandatory, this also cuts off the air supply.

(9) **Power switch**: type latched action; the feeding power is 220V to the machine. Power on or (**Power**):

```
\[ \text{Power off or (Power)} \]
```

(10) **Safety switch** (SS): type momentary action, is normally open, it is used to protect the worker if he opens the protection guard (figure 10, the transparent door of the machine), the SS will be opened.
When we close the protection guard, then SS is closed then the machine works, the SS is a main condition to stop or to work the machine.

\[ \text{SS : safety switch is closed 11 COM, 12 NO, NC is not connected to the circuit)} \]

SS: Safety switch is open 11 12 the machine cannot work.

(11) **Left switch end of cycle** (LSEC): type momentary action, is normally open, it is used to
indicate the end of a cycle, when the left cylinder end of stroke ledges (see mechanical components) hit this switch; the switch will close (b2) and the current passes through the circuit. If the number of cables arrive to zero (Nb=0) this mean the end of all cycles, then the switch will be closed by the left cylinder end of stroke ledges until we press again the Preset work piece counter or give a Number ≠ 0 or press the Manual cycle button, if we don’t give a Number ≠ 0 the machine will not work, if we give a Number ≠ 0, the machine will work normally. When the LSEC and RSEC are closed, the command on valves is from EV2A, in the case of open switches the command goes to EV2R.

(12) Right switch end of cycle (RSEC): type momentary action, is normally open, it is used to indicate the end of a cycle, when the right cylinder end of stroke ledges (see mechanical components) hit this switch; the switch will close (b3) and the current pass through the circuit.

Normally LSEC and RSEC work in the same time, they close and open in the same time.

![Figure 12, detailed circuits for the used push buttons.](image)

Where,
- C2P7 is going to circuit 2.
- C2P6 is going to circuit 2.
- C3P9 is going to circuit 3 which regulate the speed of the motor.
- C3P8 is going to circuit 3 which regulate the speed of the motor.
- C2P21 is going to circuit 2.
- C2P36 is going to circuit 2.
- C2P19 is going to circuit 2.
- C2P3 is going to circuit 2.
- C2P10 is going to circuit 2.
- C2P5 is going to circuit 2.
- C1P12 is going to circuit 1.
- C1P11 is going to circuit 1.
- C2P23 is going to circuit 2.
- C2P24 is going to circuit 2.
- C2P7 is going to circuit 2.

### 3.2 Programming with PLC

The reader may find difficulties to understand the mentioned symbols and circuits, well this is not important, because to understand the machine KORMA may take several weeks. The idea of
this paper is just to mention that the machine is improved using a PLC or an MLC. The PLC and MLC can be applied to any machine or equipment which will improve its functioning.

The KORMAK machine uses mechanical, pneumatic, electrical and electronic circuits to do the complete job. But the disadvantage is that the two parts of the machine are not synchronized (the main part and the second part which is a motor that supplies wire to the first part, refer to figure 9), by using the PLC, the two parts are synchronized and moreover the PLC improves the performance of the machine and replaces some circuits.

The used PLC is made in China by the company EASY ELECTRONIC Co. Ltd.

3.2.1 Input/Output of the PLC:
Inputs and outputs of the PLC are:

Input:  
X0  
X1  
X2

Output:  
Y2  
Y3

Symbol tables:
**Input symbol:**
- X0 = SB*SS  
- X1 = OB  
- X2 = (R70C1*ACB+MCB)*RSEC*LSEC

**Output symbol:**
- Y2 = R3  
- Y3 = 1+R5C1*R4  
- Y4 = EV1  
- Y5 = 1+R2*EV2CS*PWPC-R5*R50  
- Y6 = EV2IP*R51  
- Y7 = 1+MSS2  
- Y10 = 1

3.2.2 Grafcet level 1
Grafcet level 1 is presented in the figure 13.
Figure 1, Grafcet level 1.

3.2.3 Grafcet level 2

Grafcet level 2 is presented in the figure 14.
3.2.4 Program List

The written program using PLC in this subsection has 84 lines.

BEGIN THE PROGRAM

INITIAL STATE S0

Active the State S0

READ CONDITIONS X0 AND X1

LD X0
AND X1
AND X2

If conditions are satisfied (X0=1 AND X1=1 AND X2=1) Set the State S2

SET S2

STG S3

Active the State S3

Actions or outputs are inactive (RST=reset) when S3 is on State

RST Y2
RST Y3
RST Y4
RST Y5
RST Y7
RST Y10

JUMP function is like GO TO State S0

JUMP S0

Active the State S2

STG S2

Actions or outputs are active when S2 is on State

SET Y2
SET Y3
SET Y4
SET Y5
SET Y7
SET Y10

READ CONDITIONS X0 OR X1

LDI X0
ORI X0

If conditions are satisfied (X0=0 OR X1=0) Set the State S3

SET S3

READ CONDITIONS X0 AND X1 AND X2

Figure 14, Grafcet level 2.
3.2.5 Ladder program

This section presents the same program written using the same PLC, it takes approximately 5 pages to be completed. The same Ladder will be written using the MLC, and it will be clear which one is much easier to be completed.

This program presents the functioning of the Grafcet Level 1 and 2.

The following subsection presents the Ladder Program of the same Grafcet level 1 and 2 using the same PLC (EASY ELECTRONIC).
BEGIN THE PROGRAM

INITIAL STATE S0

active the State S0

READ CONDITIONS X0 AND X1

LD - Load; AND - and Load

if conditions are satisfied Set the State S1

active the State S1

actions or outputs are active when S1 is on State
READ CONDITIONS X0 OR X1

LDI = Load inverse; ORI = OR Load inverse

if conditions are satisfied (X0=0 OR X1=0) Set the State S3

READ CONDITIONS X0 AND X1 AND X2

if conditions are satisfied (X0=1 AND X1=1 AND X2=1) Set the State S2
active the State S3

actions or outputs are inactive (RST=reset) when S3 is on State

\[
\begin{align*}
Y_2 & \quad (R) \\
Y_3 & \quad (R) \\
Y_4 & \quad (R) \\
Y_5 & \quad (R) \\
Y_7 & \quad (R) \\
Y_{10} & \quad (R)
\end{align*}
\]

JUMP function is like GO TO State S0

active the State S2

actions or outputs are active when S2 is on State

\[
\begin{align*}
Y_2 & \quad (S)
\end{align*}
\]
READ CONDITIONS X0 OR X1

if conditions are satisfied (X0=0 OR X1=0) Set the State S4

READ CONDITIONS X0 AND X1 ORI X2

if conditions are satisfied (X0=1 AND X1=1 OR X2=0) Set the State S5
3.2.6 PLC Connection inputs and outputs

Figure 15 represents the connections of the circuits with the inputs and outputs of the PLC.
Figure 15, Connection of the machine to the PLC.

**Input connection only:**
Figure 16 represents the connections of circuits with the inputs of the PLC.
Output connection only:

Figure 17 represents the connections of circuits with the outputs of the PLC.

3.2.7 Diagram circuit

The Diagram circuit of the PLC of the previous connections is presented in the figure 18.
3.3 Programming with MLC

3.3.1 Grafcet level 2

The Grafcet of the level 2 which is presented in figure 14 programmed with the PLC can be simplified using MLC as in the figure 19.

![Diagram of the connection with the PLC.](image)

**Figure 18**: Diagram of the connection with the PLC.

**Figure 19**: Grafcet level 2 using MLC.

It can be written also using names of the inputs and outputs of the MLC such as in figure 20.

![Grafcet level 2 using the names of relays and inputs of the MLC.](image)

**Figure 20**: Grafcet level 2 using the names of relays and inputs of the MLC.

Where,

- X0=I1;
- X1=I2;
- X2=I3
- Y2=R1;
- Y3=R2;
- etc.

If one compares figure 19 which is the same program as the one programmed using PLC in figure 14, it is very clear that the MLC is much simpler in programming than the PLC for some applications.
3.3.2 Ladder program

![Ladder program diagram]

Figure 21: Ladder program using the names of relays, inputs and outputs of the MLC.

Where,
- I1: first input.
- I2: second input.
- RI1P4: intermediate relay R1 coil controlled by a 12 VDC through pin 4.
- RI1C1: first contact of the intermediate relay R1 with pins (3 and 2).
- RI2P4: intermediate relay R2 coil controlled by a 12 VDC through pin 4.
- RI2C1: first contact of the intermediate relay R2 with pins (3 and 2).
- R1: coil of the output relay R1.
- R2: coil of the output relay R2.

The Ladder program of the PLC is approximately 5 pages, and for the MLC is 4 lines, thus it is very clear that the MLC is much easier in programming using Ladder program.

3.3.3 Connection of the Inputs and intermediate relays of the MLC

Figure 22 represents the connections of the MLC which give the same function of the PLC presented in the previous sections.

The connections of the MLC are in green color, this is called the manual programming because it gives the same functionality of the PLC, but instead of programming it using a computer, it is programmed using wires. For example in the case of this paper, to program using MLC, one needs only 7 wires connected as in figure 22 in green color to give the same functionality of the PLC described in the section 3.2.4. One can remark that the written program using PLC in the subsection 3.2.4 has 84 lines which take about half an hour only to be written, while for the MLC, it needs only 7 wires that can be connected within one minute.

3.3.4 Multi-Head Wire

A Multi-Head Wire is a wire that has many heads as presented in figures 23.a and 23.b.
This kind of wires is very useful for the MLC, it facilitates the programming part by simple connections, when the same Voltage is required for many connections at the same time. This kind of wires is proposed in this paper in purpose to be used with the MLC. It can be also used for other applications.

4 CONCLUSION AND FUTURE WORK

4.1 Conclusion

This paper presents a new circuit and invention called MLC (Manual Logic Controller) which can replace the well known PLC (Programmable Logic Controller) for many applications; it has much more advantages compared to the PLC, and its programming is much simpler as stated in this paper.

The studied case in this paper is a real case when the author improved the performance of an old machine KORMAK TS 33 which cuts and strips wires, these kinds of machines are very important in the fabrication of lighting fixtures especially when fast production is required. The author developed two methods of programming, the first one using a PLC, and the second one using an MLC. A comparison between these two methods is presented, and it is clear that the MLC is much better than the PLC for such applications, especially for small and medium programs are required. It is very clear that the Grafcet and Ladder parts using MLC are much simpler than the PLC; therefore, MLC takes advantages ahead the PLC for small and medium applications.

It is important to note that the MLC can be much more complicated than the presented one in this paper; it can have digital timers, analog inputs, etc. But the main idea in this paper is to program manually the MLC using wires instead of using a computer. When a complex functionality is required, it is better to use a PLC.

4.2 Future work

In conclusion, the author encourages all engineers and manufacturers to use the MLC instead of the PLC for simple and medium applications where a few number of operations are required. The only disadvantage of the MLC is that it can’t be used for very complex programs; otherwise all the advantages are presented in it. The future work can be presented by developing specific timers that can improve the performance of the MLC, producing specific wires which have many heads as presented in figure 23, they will facilitate the programming. And finally, the main purpose of any papers, devices or inventions is to facilitate the utilization, and simplify the life, thus, the PLC and MLC can be improved according to the utilization in the simplest manner which has a low cost and high efficiency.

Figure 23: A Multi-Head Wire.
REFERENCES


Spatial Spectrum Utilization Efficiency Metric for Spectrum Sharing System

A. A. Ayeni¹, N. Faruk¹, N. T. Surajudeen-Bakinde², R. A. Okanlawon¹ and Y. A. Adediran²
University of Ilorin
¹Department of Telecommunication Science
²Department of Electrical and Electronics Engineering
ayeni.a@unilorin.edu.ng, faruk.n@unilorin.edu.ng, yinusaadediran@gmail.com and deenmat@unilorin.edu.ng

ABSTRACT

Increasing demand for spectrum in recent years has led to a greater emphasis on spectrum efficiency. A metric is needed to efficiently evaluate the utilization of the spectrum in a spectrum sharing system. In this work, we have introduced a new metric, called spatial spectrum utilization efficiency (SSUE) for evaluating the efficiency of reusing the TV spectrum. The model employed in this work, consists of a single primary transmitter network and multiple secondary networks transmitting outside the noise limited grade B contour. Simulation results indicate that for a typical low power TV transmitter of 1 kW and 4 W secondary users, the minimum and maximum achievable spectrum efficiency limits for primary networks are 28% and 54%, when the PU height are 100 m and 600 m respectively. However, the efficiency could be increased up to about 83% when 2500 nodes (i.e. SU’s) are deployed. Also, the sensing-only method does not utilize the spectrum as efficiently as the proposed method, due to the large protection margins. However, the proposed method has prior knowledge of the specific interference protection requirements for each primary network which maximizes utilization of the spectrum.

KEYWORDS

Spectrum utilization efficiency; spatial spectrum efficiency; frequency reuse; cognitive radio; metric

1 INTRODUCTION

Demand for wireless communication technologies and systems keeps increasing as reported in Cisco network index [1], and this has reached a stage whereby the capacity can only be achieved by improving spectrum utilization. Radio spectrum is found to be scarce, expensive and limited. Essentially, wireless communication systems have become part of our lives as we use them in our daily activities. Also, as stated in [2], the exponential increase in the use of high powered wireless devices and the spectrum availability or rather, the lack of it, is a major challenge. Dynamic spectrum access (DSA) vision has centered on the concepts of spectrum sharing between licensed and unlicensed users. The main goal of DSA is to create flexibility in the spectrum usage so that secondary users could have access to the spectrum and this technique is expected to increase spectrum utilization.

In recent years, there has been greater emphasis on the efficiency with which spectrum is used. The existing metrics for evaluating spectrum efficiency are not directly applicable to spatial domain, apart from cellular system. With the incidence of digital switch-over, secondary networks are expected to share the cleared and/or interleaved frequencies, thus the necessity to have a suitable metric that could be used to evaluate the spatial utilization efficiency of the system.

In this work, we have introduced a new metric, called spatial spectrum utilization efficiency (SSUE) for evaluating the efficiency of reusing spectrum in cognitive radio networks. The work investigates the performance of the metrics as a function of primary and secondary user system parameters, such as the height, transmit power, gain and frequency. The impact of the regulatory sensing rule, interference protection margin and efficiency was also investigated.

The rest of the paper is organized as follows: Section II provides related work; Section III presents the system model; path loss measurements and the model used are presented in Section IV; simulation results are presented in
Section V; and finally; Section VI concludes the paper.

2 RELATED WORKS

Several studies have sought to develop a metric, for evaluating the efficiency and utilization for different systems. The Spectrum Efficiency Working Group [3], in 2002, described the term efficiency to be commonly used to relate how much output can be produced, relative to a certain amount of input, usually expressed in percentage. However, Marcus et al. [4] classified the efficiency of spectrum use as technical, economic and social efficiency. Gurney et al. [5] examined the differences in spectrum utilization between sensing-only and geo-location enabled devices. The work concludes that geo-location devices efficiently utilize the spectrum better than the sensing only method. Spectrum is viewed in [6] as a physical resource which is used exclusively or denied other users to avoid interference. The Commerce Spectrum Management Advisory Committee [7] stated that the simplest way to increase efficiency, in a global sense, is to locate unused or severely underused frequency bands and reassign them to services that will actually utilize them. This concept embraces the DSA vision. In [8] spectrum efficiency is viewed as in [4]; however, the social aspect was not considered. Rather, functional efficiency was considered. Zhang et al [9] has introduced area spectral efficiency (ASE) as a metric to quantify the spectral utilization efficiency of cellular systems. The work takes spatial spectrum perspectives into consideration.

Tao Chen et al. [10] proposed energy efficiency metrics for green wireless communications. The work provides a holistic view from system architecture to component level. In [11], a time-slotted spectrum sharing protocol, named Channel Usage and Collision Based MAC protocol (CUCB-MAC), was proposed. It is a modification of a MAC protocol, named Collision-Based MAC protocol (CB-MAC), which depends on three criteria for the allocation of channels to the SUs; 1) counting collisions number for each SU, 2) availability probability for all the available channels and then 3) exclusion of the available channels. It was proved that using the proposed CUCB-MAC protocol outperforms the original CB-MAC protocol on all the measured performance metrics.

3 SYSTEM MODEL

In this model, a system in which a primary transmitter communicates with primary receivers within the noise-limited contour (Grade B contour) [12] is considered. The grade B contour is the service or protection area of the primary cell. The SU can reuse the same frequency near the cell border of the PU if it meets the required condition, specified by FCC for white space. The TV service area is the geographical area within the TV noise-limited grade B contour, i.e. where the received signal strength for channels 14 through 69 exceeds 64 dBu and 41 dBu for analog and DTV systems respectively [5], [13].

These contours were calculated in accordance with specific procedures described in Sections 73.622 and 73.625 of the FCC Rules and Regulations [14]. For such protection to be guaranteed, the SUs must be located outside of the grade B contour of the PU for a minimum distance (keep-out-distance) as described by FCC in [15]. We assume a primary user of height $h_p$ (m), transmitter power $P_{Tx}$ (dBm) and operating frequency $f$ (MHz), for the TV band, as shown in Fig 1. Let $r_p$ represent the protection contour-radius and $r_n$ be the no-talk radius i.e. the minimum required separation distance between the primary transmitter (PTx) and the secondary transmitter STx.

![Fig.1. Spectrum sharing for single primary transmitter and single secondary user](image-url)
Outside the protection margin, there exists a safe-to-transmit region, given by \( r_s \subseteq \mathbb{R}^3 \), in which a co-located transceiver can operate on the same frequency band. A white space in space-time is defined as an indicator function \( 1_D : \mathbb{R} \rightarrow [0,1] \) given by

\[
1_D (x, f, t) = \begin{cases} 
0 & \text{if } x \in r_s \\
1 & \text{if } x \in \mathbb{R}^3 \setminus r_s 
\end{cases}
\] (1)

where \( x \) is a coordinate in space where a secondary transmitter may or may not transmit, \( f \) is the frequency and \( t \) is the time. The received signal level at the grade B contour \( P_r \) is given by:

\[
P_r = P_T + G - \Psi(r) - N_0 - l(r_p) \geq \lambda
\] (2)

where \( G \) is the gain of the antenna (in dBi), \( N_0 \) is the thermal noise level (in dBm), \( l(r_p) \) is the path loss and \( \Psi(r) \) is the fade margin. Therefore, the range of the protection contour is derived as follows:

\[
r_p = l_p^{-1}(P_T + G - \Psi(r) - N_0 - \lambda)
\] (3)

In addition to the protection region, the secondary network can only reuse the spectrum if it is outside the grade B contour with additional distance, called the protection distance \( d_n(\Delta) \), which is dependent on the interference protection ratio, \( \Delta \), specified by the regulators. Therefore, the no-talk radius is derived as follows:

\[
r_n(\lambda, p_T, h_i, \Delta) = r_p(\lambda, p_T, h_i) + d_n(\Delta)
\] (4)

where \( d_n(\Delta) \) is defined as:

\[
d_n(\Delta) = l_p^{-1}(P_T + G - \Psi(r) - N_0 - \lambda + \Delta_i)
\] (5)

and

\[
d_n(\Delta) = r_p(\lambda, p_T, h_i, \Delta) - r_p(\lambda, p_T, h_r)
\] (6)

\( r_p(\lambda, p_T, h_r) \) is the protection radius of the TV transmitter, which depends on the system parameters and \( d_n(\Delta) \) is the additional protection distance that depends on the protection margin \( \Delta_i \).

FCC provides 23 dB as the interference protection ratio in a co-channel scenario.

Let \( A_p \) be the area of the PU cell, \( A_s \) the area of SU cell and \( A_{total} \) the total coverage area for a given geographical area (Ilorin, Kwara State, Nigeria in this case). Then, the SSUE is given by:

\[
A_p(h_{tp}, PTx, \Delta, \lambda) + A_s + A_s + A_s + \ldots + A_s(h_{tp}, STx, \tau) \times 100 \%
\] (7)

\[
SSUE = \frac{A_p(h_{tp}, PTx, \Delta, \lambda) + \sum_{l=0}^{N} A_s(STx, h_{tx}, \tau)}{A_{total}} \times 100 \%
\] (8)

where \( N \) is the total number of secondary users in the network.

### TABLE 1. PRIMARY USER SIMULATION PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary user Transmit Power, PTx</td>
<td>1 kW, 10 kW…100 kW</td>
</tr>
<tr>
<td>Center frequency, f</td>
<td>CH 35 (583.25 MHz)</td>
</tr>
<tr>
<td>Primary user Height, Hp</td>
<td>100,200, 300….600 m</td>
</tr>
<tr>
<td>Height of receiver, Rx</td>
<td>10 m</td>
</tr>
<tr>
<td>Field strength at grade B contour</td>
<td>41 dBu</td>
</tr>
<tr>
<td>Interference ratio, ( \Delta )</td>
<td>23 dB</td>
</tr>
</tbody>
</table>
### TABLE 2. SECONDARY USER SIMULATION PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secondary user Transmit power, STx</td>
<td>100 mW, 1 W, 2 W, 3 W, 4 W</td>
</tr>
<tr>
<td>Channel Spacing</td>
<td>8 MHz</td>
</tr>
<tr>
<td>Secondary user Height, Hs</td>
<td>30 m</td>
</tr>
<tr>
<td>Height of Receiver, Hr</td>
<td>1.5 m</td>
</tr>
<tr>
<td>Sensitivity level, ( \tau )</td>
<td>-85 dBm</td>
</tr>
<tr>
<td>Antenna gain, G</td>
<td>10 dBi</td>
</tr>
<tr>
<td>Thermal noise</td>
<td>-105 dBm</td>
</tr>
<tr>
<td>Receiver noise figure</td>
<td>10 dB</td>
</tr>
<tr>
<td>Noise power (N)</td>
<td>-95 dBm</td>
</tr>
</tbody>
</table>

### 4 PATH LOSS MEASUREMENTS

It is noted in [16], [17], [18] and many other published research that path loss model plays a key role in coverage estimation and can have tremendous impact on the performance of a system. Particularly, the empirical models which are environment-dependent can have significant errors in path loss prediction when they are employed in a different environment other than the one for which they were initially built. The errors were apparent when 10 models were tested in our case study area as reported in [19], [20]. This necessitates our modifying the existing prediction model, i.e. Hata-Davidson, which had shown better results in our earlier studies with the results of the optimization published in [21]. This would ensure that errors are significantly minimized and, thus consequently, protect the primary users from excessive interference from secondary users, hence maximizing the local spectrum usage.

In this study, a large scale field strength measurements in the VHF and UHF frequencies, along six different routes, that spanned through the urban, suburban and rural areas in Kwara state, Nigeria, were carried out, using a dedicated Agilent spectrum analyzer, N9342C (100 Hz-7 GHz) with a displayed average noise level of -164 dBm/Hz. Analysis of data sets from the field measurements were carried out and the result used to build an optimized model, thereby predicting the service coverage and consequently, the protection margins for the primary transmitters.

### 5 SIMULATION AND RESULTS

Fig. 3 shows the coverage distance for secondary network as a function of transmit power. It is however expected that the coverage range would increase with increase in transmit power. In this scenario, we assume a worst case situation, in which -85 dBm is considered as the receive threshold, which is far above the -93 dBm considered as the threshold for TVWS devices [22]. Simulation parameters used for both primary and secondary networks are shown in Tables 1 and 2.

Fig. 4 shows the variation of the field strength with distance for the Kwara State TV station (KWTV), transmitting on UHF channel (CH) 35 (i.e. 583.25 MHz) at 1 kW power, as a function of height. The 41 dBu ITU regulation for service contour, in
digital TV systems, is considered. It is observed that the grade B contour is 22 km, for a 100 m antenna and this increases to 28 km, 31 km, 34 km, 37 km and 40 km for 200 m, 300 m, 400 m, 500 m and 600 m antenna heights respectively.

Fig. 5 shows the spatial spectrum efficiency for cognitive radio system as a function of the number of secondary users and the height of the PtX. In this scenario, a 1 kW primary transmitter power is considered with secondary users transmitting at 4 W. This power level was considered as the maximum transmitter power for a fixed secondary user device and 100 mW for portable devices [22]. The maximum and minimum achievable efficiency values for the primary network only are 54% and 28%, when the heights are 600 m and 100 m respectively. The efficiency increases to about 60% when 500 secondary networks are deployed and 83% for 2500 SUs. This is expected to further increase, with increase in the number of secondary users. However, there is decrease in efficiency of about 36% when the height of the primary transmitter is decreased from 600 m to 100 m for a fixed 2500 secondary network.

Fig. 6 shows the impact of the primary network power on efficiency. Considering a 100 m height antenna, in the primary network only, the efficiency is 29% when the PtX is 1 kW. This increases to 50% for 10 kW and about 83% for 100 kW. It is conclusive that the transmit-power has more significant impact on the efficiency than the primary transmitter-heights do. The impact of secondary network transmit power on efficiency was also investigated as a function of the number of secondary users. A primary transmitter of 1 kW transmit power and 100 m height was considered and the result is shown in Fig. 7. As expected, the SU transmit-power affects the overall efficiency; however, the effect is more considerable when PU power is increased.

A typical scenario is when 2500 nodes are deployed with each transmitting at a maximum power of 4 W. The maximum efficiency obtained is about 60%, which could be increased to 80% (see Fig. 6) if the PU power is 10 kW, using the same SU condition of 2500 nodes and 4 W transmit power. In order to achieve 80% efficiency with low power TV transmitter, about 4250 nodes would have to be deployed, with each transmitting at full power of 4 W. This might have cost implication in real life application.

The performance of the proposed SSUE model is compared with that of the conventional sensing method. The result is shown in Fig. 8. In this
scenario, a metric called *unrecovered area* is used, together with the regulatory sensing rules. The unrecovered area is the area close to the primary user protection area where the secondary could not be recovered as a result of protection margins specified by the regulators. The total area under study $A_{\text{Total}}$ is 36,825 km$^2$.

In July 2007, the Office of Engineering and Technology in Alaska, USA, [23] released a technical report on the initial study of prototype TVWS devices, where a bench test sensing threshold of -114 dBm was proposed. However, in the phase II prototype devices laboratory test, it was found that the devices were able to detect the DTV signal with sensitivity range of -116 dBm to -126 dBm. This keeps the regulators in doubt on the benchmark threshold. The FCC had suggested -114 dBm as the criteria of the empty spaces for TVWS while Ofcom proposed -116 dBm.

From our study, the FCC’s -114 dBm provides an extra 3.5 dB protection, to the original 23 dB protection margin, proposed for a primary transmitter. In [15], it was found that the sensing rules have significant impact on the co-channel keep-out distance and this could affect the availability of white space. From Fig. 8, a 1 kW primary transmitter was used, which was only about 28% efficient. The unrecovered area was found to be lower for the proposed model when compared with the sensing-only method. However, the unrecovered area increases with an increase in primary transmitter height, this, perhaps, is expected as the height affects the co-channel keep-out distance and, consequently, affects the availability of the white space. This relation is however not true for the sensing case.
6 CONCLUSIONS

In this work, we have introduced a new metric, called spatial spectrum utilization efficiency (SSUE), for evaluating the efficiency of reusing the spectrum in cognitive radio networks. Our findings indicate that for a typical low power TV transmitter of 1 kW and 4 W secondary users the minimum and maximum achievable spectrum efficiency limits for primary network only are 28% and 54% for PU height of 100 m and 600 m respectively. However, the efficiency could be increased up to about 83% when 2500 nodes (i.e. SUs) are deployed to reuse the spectrum under regulatory rules on interference.

In terms of unrecovered area, the proposed method is found to perform better than the sensing only method. The sensing-only method does intelligent detection but does not generally utilize the spectrum as efficiently as the proposed method due to the large protection margins in incumbent detection thresholds; and this must be built into sensing-only devices. On the other hand, the proposed method has prior knowledge of the specific interference protection requirements of each primary network which could provide flexibility in choosing the appropriate protection margins and subsequently, maximize utilization of the spectrum.

7 ACKNOWLEDGEMENTS

The authors would like to express their sincere appreciation to University of Ilorin for the purchase of dedicated Agilent spectrum analyzer used for this study and also for the funding through staff development award.

8 REFERENCES


Ciphertext Diverge-Merge Scheme of Identity-Based Encryption for Cloud-Based File Transmission Service

Makoto Sato¹ and Masami Mohri² and Hiroshi Doi³ and Yoshiaki Shiraishi⁴
¹Nagoya Institute of Technology, Aichi, Japan
²Gifu University, Gifu, Japan
³Institute of Information Security, Kanagawa, Japan
⁴Kobe University, Hyogo, Japan
¹sato.makoto@nitzlab.com, ²mmohri@gifu-u.ac.jp, ³doi@iisec.ac.jp, ⁴zenmei@port.kobe-u.ac.jp

ABSTRACT

Cloud-based file transmission services have attracted attention as the way of sending files. However, not a few people have a reluctance to put their serious information on cloud servers. Encrypting and decrypting files on user’s terminals should resolve this concern. To avoid oppressing computational power of clients’ terminals, these ones should be as light as possible. In identity-based encryption (IBE), an identity (ID) such as mobile phone numbers is used as public keys, so there is no need for public key certificates. However, IBE has the key escrow problem that the private key generator (PKG) could see any message because all private keys used in decryption are generated by it. In this paper, we show how to configure cloud-based file transmission service that uses IBE and is free from the problem. In order to show that only receivers can decrypt ciphertexts, we define two attack models that either the PKG and the service providing server or the e-mail server constituting the proposed scheme collude with an adversary, and prove that our scheme is secure against passive attack in the models. Then, we compare the amount of computation of private key extraction, encryption and decryption.

KEYWORDS

Identity-based encryption, Cloud, Secure file transmission, Key escrow problem, Ciphertext division

1 INTRODUCTION

1.1 Background

Cloud computing services have many benefits such as scalability, high performance, and easy to maintain and update [1]. Above all, cloud-based file transmission services have attracted attention as the way of sending files. On the other hand, not a few people have a reluctance to put their serious files on cloud servers [2]. Some services provide server side encryption, but this encryption allows service providers to see the contents of files on servers. We therefore consider client side encryption and decryption. These ones should be as light as possible because computational power of clients’ terminals may be shared with every process on the terminals.

If public key encryption based on public key infrastructure (PKI) is used for encryption, a receiver of encrypted messages (we call him/her “receiver” for brevity) has to get a public key certificate which certifies the validity of the key and is issued by the certificate authority (CA), the trusted third party. Certificates should be managed so that anyone who wants to send encrypted messages (we call him/her “sender” for brevity) can get them easily in advance.

In identity-based encryption (IBE), an identity (ID) such as names, addresses, e-mail addresses or mobile phone numbers can be used as public keys, so there is no need for certificates as long as the validity of the ID is ensured.

1.2 Service Model of Cloud-Based File Transmission Service

A model of cloud-based file transmission service is shown in figure 1. General service [3-5] needs two servers, the service providing server and the e-mail server. The service providing server usually publishes a web site which users who want to send
files accesses. This server is operated as a cloud server by cloud service providers. The e-mail server informs receivers of file reception.

The model shown in figure 1 is without file encryption so confidentiality of files is not protected [6]. The model with file encryption by IBE is shown in figure 2. This model needs the private key generator (PKG) besides the service providing server and the e-mail server. A sender encrypts a file $F$ with a symmetric key $SK$ and generates the encrypted file $F'$. Then the sender encrypts $SK$ with receiver’s ID to generate the ciphertext $C$. The receiver decrypts $C$ with the private key $d_{ID}$ issued by the PKG. Because all private keys are generated by the PKG, this model suffers from the key escrow problem, which the PKG could decrypt any ciphertext. We considered that if a ciphertext $C$ can be transformed into two parts $C_1, C_2$ and one of these ones is kept secret from the PKG, the key escrow problem would be resolved.

Based on this concept, we define the model of cloud-based file transmission service (see figure 3) that is free from the key escrow problem. Each entity works as follows:

1. A sender encrypts a file with a symmetric key $SK$ to generate the encrypted file $F'$, and encrypts $SK$ with receiver’s ID to generate the ciphertext $C$. Then, the sender transforms $C$ into two parts $C_1, C_2$. Finally, the sender transmits $F'$ and $C_1$ to the service providing server over a public channel and $C_2$ to the e-mail server over a secure channel.
2. The service providing server sends its URL to the e-mail server.
3. The e-mail server sends $C_2$ and the URL to the receiver over a secure channel.
4. When the receiver gets $C_2$ and the URL, he sends a file request to the service providing server.
5. The service providing server transmits $F'$ and $C_1$ to the receiver.
6. The receiver sends his ID to the PKG. The PKG authenticates him.
7. PKG generates the private key $d_{ID}$ corresponding to the ID, and send it to the receiver over a secure channel such as SSL.
8. The receiver decrypts $C$ to generate $SK$, and he gets the file $F$ by decrypting $F'$ with $SK$.

In the step (1) and (3), we assume that the whole channel through which $C_2$ is sent is a secure one. This channel can be provided as following two cases under the condition that both channels between a sender and the e-mail server and the e-mail server and a receiver are secure ones using SSL/TLS:

I. Senders and receivers use the same e-mail server.
II. Senders and receivers use different e-mail servers, X and Y, and a channel between X and Y is a secure one.
In the case I, \( C_2 \) is totally kept secret from the PKG, unless the e-mail server leaks \( C_2 \) deliberately.

In the case II, for instance, by using such as STARTTLS for SMTP [7] on the channel between X and Y, the confidentiality of \( C_2 \) can be protected.

1.3 Related Works

IBE has the key escrow problem that the PKG could see any message because all private keys used in decryption are generated by it. Some schemes resolving this problem have been proposed [8-9].

Kate and Goldberg’s scheme [8] was proposed in 2010. The scheme applies threshold techniques for PKG’s master secret key, and manages the key by multiple PKGs. PKGs not more than threshold cannot generate private keys and the scheme can be applied to three IBEs, Boneh and Franklin’s scheme [10], Sakai and Kasahara’s scheme [11] and Boneh and Boyen’s scheme [12]. In schemes using threshold techniques, receivers should communicate with multiple PKGs and be authenticated by each of them in order to gain their private keys.

Al-Riyami and Paterson’s certificateless public key encryption scheme [9] was proposed in 2003. In the scheme, private keys are generated from receiver’s secret values and partial private keys generated by the PKG, so the PKG could not see messages because it does not know receiver’s secret values. Receivers should communicate with the only one PKG so as to generate private keys. However, in this scheme, senders should communicate with receivers to get their public keys, and encryption is relatively heavy.

1.4 Our Contribution

We show how to configure cloud-based file transmission service that uses IBE and is free from the key escrow problem. As a concrete example of IBE, we cover Boneh and Franklin’s BasicIdent scheme [10]. The algorithms themselves need no revision. A ciphertext is transformed into two parts \( C_1, C_2 \) and one of these ones is kept secret from the PKG. Ciphertexts diverge at senders and merge at receivers. We call this method ciphertext diverge-merge scheme (CDMS).

Our scheme satisfies following two properties: 1) only receivers can decrypt ciphertexts. 2) the amount of calculation computed on user’s terminals is less than that of existing schemes. In order to show our scheme has the former property, we prove that our scheme is secure against passive attack even if either the PKG and the service providing server or the e-mail server collude with an adversary. Then, to show the latter property, we compare the amount of computation of private key extraction, encryption and decryption.

2 PRELIMINARIES

2.1 Bilinear Maps

Let \( \mathbb{G}_1, \mathbb{G}_2 \) be two cyclic groups of prime order \( q \). An admissible bilinear map \( \hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2 \) satisfies following properties in arbitrary \( P, Q \in \mathbb{G}_1, a, b \in \mathbb{Z}_q^* \):

- **Bilinear**: \( \hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab} \).
- **Non-degenerate**: If \( \hat{e}(P, Q) = 1, P = 0 \) or \( Q = 0 \).
- **Computable**: There are efficient algorithms to compute \( \hat{e}(P, Q) \).

2.2 Boneh and Franklin’s BasicIdent Scheme

Boneh and Franklin’s identity-based encryption scheme [10] was proposed in 2001. There are two schemes. One has been proved to be IND-ID-CPA secure and the other proved to be IND-ID-CCA secure. These schemes are standardized in RFC5091 [13]. Our scheme is based on the former scheme, BasicIdent. BasicIdent is composed of the following four algorithms.

**Setup**: Given a security parameter \( k \in \mathbb{Z}^+ \), the algorithm works as follows:

1. **Step1**: Output two groups \( \mathbb{G}_1, \mathbb{G}_2 \) of prime order \( q \), and an admissible bilinear map \( \hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2 \), and choose a random generator \( P \in \mathbb{G}_1 \).
2. **Step2**: Pick a random \( s \in \mathbb{Z}_q^* \) and compute \( P_{pub} = sP \).
Step3: Choose cryptographic hash functions $H_1: \{0,1\}^* \rightarrow \mathbb{G}_1^*$, $H_2: \mathbb{G}_2 \rightarrow \{0,1\}^n$. $\mathbb{G}_1^*$ denotes the set $\mathbb{G}_1 \setminus \{O\}$, where $O$ is the identity element in the group $\mathbb{G}_1$. The message space is $\mathcal{M} = \{0,1\}^n$ and the ciphertext space is $\mathcal{C} = \mathbb{G}_1^* \times \{0,1\}^n$. The system parameters are $\text{params} = (< q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, n, P, P_{pub}, H_1, H_2 >)$. The master key is $msk = s$.

**Extract**: For a given string $ID \in \{0,1\}^*$ the algorithm does: 1) Compute $Q_{ID} = H_1(ID) \in \mathbb{G}_1^*$. 2) Set $d_{ID} = sQ_{ID}$ as the private key corresponding to the ID.

**Encrypt**: To encrypt $M \in \{0,1\}^n$ with the identity $ID \in \{0,1\}^*$, do the following: 1) Compute $Q_{ID} = H_1(ID) \in \mathbb{G}_1^*$. 2) Pick a random $r \in \mathbb{Z}_q^*$ and set the ciphertext to be $C = < rP, M \oplus H_2(\hat{e}(Q_{ID}, P_{pub})^r) >$.

**Decrypt**: Let $C = < U, V > \in \mathcal{C}$ be a ciphertext encrypted with the identity ID. To decrypt $C$ with the private key $d_{ID}$ compute: $V \oplus H_2(\hat{e}(d_{ID}, U)) = M$.

We say that an identity-based encryption scheme is semantically secure (IND-ID-CPA) if no polynomially bounded adversary $A$ has a non-negligible advantage against the Challenger in the following IND-ID-CPA game:

**Setup**: The challenger takes a security parameter $k$ and runs the Setup algorithm. It gives the adversary the system parameters $\text{params}$. It keeps the master secret key to itself.

**Phase 1**: The adversary issues private key extraction queries $ID_1, \ldots, ID_m$. The challenger responds by running algorithm Extract to generate the private key $d_i$ corresponding to the identity $ID_i$. It sends $d_i$ to the adversary. These queries could be asked adaptively.

**Challenge**: Once the adversary decides that Phase 1 is over, it outputs two equal length messages $M_0, M_1 \in \mathcal{M}$ and an identity $ID^*$ on which it wishes to be challenged. The only constraint is that $ID^*$ did not appear in any private key extraction query in Phase 1. The challenger chooses a random bit $b \in \{0,1\}$ and sets $C = \text{Encrypt}(\text{params}, ID^*, M_b)$. It sends $C$ as the challenge to the adversary.

**Phase 2**: The adversary issues more extraction queries $ID_{m+1}, \ldots, ID_n$. The only constraint is that $ID_1 \neq ID^*$. The challenger responds as in Phase 1.

**Guess**: Finally, the adversary outputs a guess $b' \in \{0,1\}$ and wins the game if $b = b'$.

### 3 OUR SCHEME

#### 3.1 Ciphertext Diverge-Merge Scheme

In this subsection, we show our scheme named ciphertext diverge-merge scheme (CDMS) for IBE.

We choose Boneh and Franklin’s BasicIdent scheme as a concrete example of IBE. CDMS is based on the assumption that if a ciphertext $C$ can be transformed into two parts $C_1, C_2$ and one of these ones is kept secret from the PKG, the key escrow problem would be resolved. The algorithms themselves need no change but a ciphertext is sent as follows (we show the flow of sending a ciphertext in figure 4):

**CDMS-Encrypt**: According to BasicIdent scheme’s Encrypt, a sender generates the ciphertext $C = < C_1, C_2 > = < rP, M \oplus H_2(\hat{e}(Q_{ID}, P_{pub})^r) >$. Then $C$ diverges, that is, he sends $C_1$ to the server A over a public channel and $C_2$ to the server B over a secure channel.

**CDMS-Decrypt**: the server B sends $C_2$ to the receiver over a secure channel and the server A sends $C_1$ to the receiver over a public channel. Then $C_1$ and $C_2$ merge, that is, the receiver decrypts the ciphertext $C = < C_1, C_2 >$ according to BasicIdent scheme’s Decrypt.

Note that the service providing server and the e-mail server described in section 1 are considered as the server A and the server B, respectively.

### 3.2 Attack Models

We define security requirements of our scheme as “Even if either the PKG and the service providing server or the e-mail server collude with an
adversary, our scheme is secure against passive attack”. These requirements are based on the assumption that cloud servers are “honest-but-curious” [14]. This means cloud servers will follow our proposed protocol in general, but try to find out as much secret information as possible.

According to security requirements, we evaluate following two attack models:

- Attack model 1: the PKG and the service providing server collude with an adversary
- Attack model 2: the e-mail server colludes with an adversary

We prove that the proposed scheme satisfies following security against two attack models:

- Attack model 1: CDMS-IND-CPA secure
- Attack model 2: CDMS-IND-ID-CPA secure

If the PKG colludes with an adversary, the adversary can get private keys corresponding to IDs. Therefore, in the attack model 1, we aim at CDMS-IND-CPA secure instead of CDMS-IND-ID-CPA secure.

### 3.2.1 Attack Model 1 and Security in the model

In this model, we assume that the PKG and the service providing server collude with an adversary. Security in this model is defined with the following game:

**Setup**: The challenger takes a security parameter \( k \) and runs **Setup** algorithm, then it gives an adversary **params** and **msk**.

**Phase 1**: The adversary issues the following query several times.

- **\( C_1 \)** query: When the adversary send a message \( M \) and ID\(_i\) to the challenger, the challenger responds by running **Encrypt** algorithm to generate the ciphertext \( C = < C_1, C_2 > \). Then, the challenger sends \( C_1 \) to the adversary.

**Challenge**: Once the adversary decides that **Phase 1** is over, it outputs two equal length messages \( M_0, M_1 \) and an identity ID on which it wishes to be challenged. The only constraint is that ID did not appear in any query in **Phase 1**. The challenger chooses a random \( \sigma \in \{0,1\} \) and sets \( C^* = < C_1^*, C_2^* > = \text{Encrypt}(\text{params}, \text{ID}, M_\sigma) \). It sends \( C_1^* \) as the challenge to the adversary.

**Phase 2**: The adversary issues the same query in **Phase 1**. The only constraint is that ID\(_i\) \( \neq \text{ID} \).

**Guess**: Finally, the adversary outputs a guess \( \sigma' \in \{0,1\} \) and wins the game if \( \sigma = \sigma' \). If the probability that the adversary wins the game is \( 1/2 + \epsilon(k) \), the adversary has the advantage \( \epsilon(k) \).

**Definition 1** We say that our scheme is CDMS-IND-CPA secure, if no polynomial time adversary has a non-negligible advantage \( \epsilon(k) \) against the challenger.

### 3.2.2 Attack Model 2 and Security in the model

In this model, we assume that the e-mail server colludes with an adversary. Security in this model is defined with the following game:

**Setup**: The challenger runs **Setup**, then it gives an adversary **params**.

**Phase 1**: The adversary can issue the following query:

- **Extraction query**: The adversary sends ID\(_i\) to the challenger, then the challenger responds by running algorithm **Extract** to generate the private key \( d_{\text{ID}_i} \) corresponding to the identity ID\(_i\). It sends \( d_{\text{ID}_i} \) to the adversary.
- **\( C_2 \)** query: When the adversary send a message \( M \) and ID\(_i\) to the challenger, the challenger responds by running **Encrypt** algorithm to generate the ciphertext \( C = < C_1, C_2 > \). Then, the challenger sends \( C_2 \) to the adversary.

**Challenge**: Once the adversary decides that **Phase 1** is over, it outputs two equal length messages \( M_0, M_1 \) and an identity ID on which it wishes to be challenged. There is a constraint that ID did not appear in any extraction query in **Phase 1**. The challenger chooses a random \( \sigma \in \{0,1\} \) and
sets \( C^* = \langle C_1^*, C_2^* \rangle = \text{Enc}(\text{params}, M_\sigma, \text{ID}) \). It sends \( C_2^* \) as the challenge to the adversary.

**Phase 2:** The adversary can issue the same query in **Phase 1**. The only constraint is that \( \text{ID}_i \neq \text{ID} \).

**Guess:** Finally, the adversary outputs a guess \( \sigma' \in \{0,1\} \) and wins the game if \( \sigma = \sigma' \). If the probability that the adversary wins the game is \( 1/2 + \varepsilon(k) \), the adversary has the advantage \( \varepsilon(k) \).

**Definition 2** We say that our scheme is CDMS-IND-ID-CPA secure, if no polynomial time adversary has a non-negligible advantage \( \varepsilon(k) \) against the challenger.

### 4 Security Proof

We prove the security of the proposed scheme in the attack models defined in the subsection 3.2.

#### 4.1 Security Proof in the Attack Model 1

**Theorem 1** Suppose Boneh and Franklin’s BasicIdent scheme is IND-ID-CPA secure, our scheme is CDMS-IND-CPA secure in the attack model 1 in the random oracle model.

**Proof** Let \( A \) be an CDMS-IND-CPA adversary with advantage \( \varepsilon(k) \) against our scheme. We prove that there is an adversary \( B \) which has advantage at least \( \varepsilon(k) \) against Boneh and Franklin’s BasicIdent scheme simulator (Given input, it responds according to algorithms of the BasicIdent scheme). \( B \) works by interacting with \( A \) in the CDMS-IND-CPA game as follows:

**Setup:** The adversary \( B \) receives \( \text{params} \) and \( \text{msk} \) from the simulator. It gives them to the adversary \( A \).

**Phase 1:** The adversary \( A \) issues the following query several times.

\( C_1 \) query. When the adversary \( A \) send a message \( M \) and \( \text{ID}_i \) to the adversary \( B \), the adversary \( B \) sends them to the simulator. Then, the simulator generates the ciphertext \( C = \langle C_1, C_2 \rangle \) by running \text{Encrypt} algorithm and sends it to the adversary \( B \).

The adversary \( B \) picks \( C_1 \) and sends it to the adversary \( A \).

**Challenge:** The adversary \( A \) sends \( \langle \text{ID}, M_0, M_1 \rangle \) to the adversary \( B \), then \( B \) sends the simulator them. The simulator picks a random \( \sigma \in \{0,1\} \), and sets \( C^* = \langle C_1^*, C_2^* \rangle = \text{Encrypt}(\text{params}, M_\sigma, \text{ID}) \). It sends the adversary \( B \) \( C^* \). Finally, the adversary \( B \) sends the adversary \( A \) \( C_1^* \).

**Phase 2:** The adversary \( A \) issues the same query in **Phase 1**. The only constraint is that \( \text{ID}_i \neq \text{ID} \).

**Guess:** Finally, the adversary \( A \) outputs \( \sigma' \in \{0,1\} \). The adversary \( B \) outputs \( \sigma' \) as its guess.

The simulation above shows there is an adversary \( B \) that has advantage at least \( \varepsilon(k) \) against Boneh and Franklin’s BasicIdent scheme simulator if there is an adversary \( A \) that has advantage \( \varepsilon(k) \) against our scheme.

#### 4.2 Security Proof in the Attack Model 2

**Theorem 2** Suppose Boneh and Franklin’s BasicIdent scheme is IND-ID-CPA secure, our scheme is CDMS-IND-ID-CPA secure in the attack model 2 in the random oracle model.

**Proof** Let \( A \) be an CDMS-IND-ID-CPA adversary with advantage \( \varepsilon(k) \) against our scheme. We prove that there is an adversary \( B \) which has advantage at least \( \varepsilon(k) \) against Boneh and Franklin’s BasicIdent scheme simulator. \( B \) works by interacting with \( A \) in the CDMS-IND-ID-CPA game as follows:

**Setup:** The adversary \( B \) receives \( \text{params} \) from the simulator. It gives \( \text{params} \) to the adversary \( A \).

**Phase 1:** The adversary \( A \) issues the following query several times.

\( C_2 \) query. When the adversary \( A \) send a message \( M \) and \( \text{ID}_i \) to the adversary \( B \), the adversary \( B \) sends them to the simulator. Then, the simulator generates the ciphertext \( C = \langle C_1, C_2 \rangle \) by running \text{Encrypt} algorithm and sends it to the adversary \( B \). The adversary picks \( C_2 \) and sends it to the adversary \( A \).
Extraction query: The adversary $A$ sends $ID_i$ to
the adversary $B$, then the adversary $B$
receives the private key $d_{ID_i}$ corresponding
to the identity $ID_i$ from the simulator, then it
gives the adversary $A$ $d_{ID_i}$.

**Challenge:** The adversary $A$ sends $<ID, M_0, M_1>$
to the adversary $B$, then $B$ sends the simulator
to them. The simulator picks a random $\sigma \in \{0,1\}$, and
sets $C^* = <C_1^*, C_2^*> = Encrypt(params, M_\sigma, ID)$. It sends the adversary $B$ $C^*$. Finally, the
adversary $B$ sends the adversary $A$ $C_2^*$.

**Phase 2:** The adversary $A$ issues the same query in
**Phase 1.** The only constraint is that $ID_i \neq ID$.

**Guess:** Finally, the adversary $A$ outputs $\sigma' \in \{0,1\}$.
The adversary $B$ outputs $\sigma'$ as its guess.

The simulation above shows there is an adversary $B$
that has advantage at least $\varepsilon(k)$ against Boneh
and Franklin’s BasicIdent scheme simulator if
there is an adversary $A$ that has advantage $\varepsilon(k)$
amongst our scheme.

5 COMPARISON OF THE COMPUTATIONAL COST

We compare the amount of computation (see
Table 1) of private key extraction, encryption and
decryption with Al-Riyami and Paterson’s scheme
[9] by the same way as [12] does. Symbols used in
the table mean:

- $P$: pairings in $\mathbb{G}_1$.
- $E_{\text{General}}$: general exponentiations in $\mathbb{G}_1$, i.e.,
sub-expressions of the form $b^x$ where $b$ is
not known ahead of time.
- $E_{\text{Fix-base}}$: exponentiations with a fixed base in
$\mathbb{G}_1$. This can be calculated much faster after
investing in a moderate amount of pre-
computation.
- $H$: hashing operations in $\mathbb{G}_1$.

$P$ has the largest calculation load and followed by
$E_{\text{General}}, E_{\text{Fix-base}}$ and $H$ in descending order.

First, as to private key extraction, the proposed
scheme needs less computation than Al-Riyami
and Paterson’s scheme by one $E_{\text{General}}$ operation.

<table>
<thead>
<tr>
<th>Table 1. Comparison of the amount of computation ($P$: pairings, $E_{\text{General}}$: general exponentiations in $\mathbb{G}<em>1$, $E</em>{\text{Fix-base}}$: exponentiations with a fixed base in $\mathbb{G}_1$, $H$: hashing operations in $\mathbb{G}_1$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The proposed scheme</td>
</tr>
<tr>
<td>----------------------</td>
</tr>
<tr>
<td><strong>Private key extraction</strong></td>
</tr>
<tr>
<td><strong>Encryption</strong></td>
</tr>
<tr>
<td><strong>Decryption</strong></td>
</tr>
</tbody>
</table>

As to encryption, our proposed scheme needs
two pairing operations less than Al-Riyami and
Paterson’s scheme.

Considering encryption is executed on client’s
terminals, the proposed scheme is reasonable in
that it needs smaller calculation load.

6 CONCLUSIONS

Instead of existing ways of transmitting files,
cloud-based file transmission services have
attracted attentions. However, not a few people
have a reluctance to put their serious information
on cloud servers [2].

In order to prevent service providers from seeing
the contents of files on servers, we assumed that
encryption and decryption are operated on the
client’s terminals. These operations should be as
light as possible because computational power of
clients’ terminals may be shared with every
process on the terminals.

Thinking of using IBE, it eliminates the need for
public key certificates which are necessary in PKI.
However, IBE has the key escrow problem that
the PKG could decrypt any ciphertext.

We therefore showed how to configure cloud-
based file transmission service with IBE that is
free from the key escrow problem. As a concrete
example of IBE, we chose Boneh and Franklin’s
BasicIdent scheme [10]. We did not make any
change to the algorithms, but devised the way of
sending ciphertexts. A ciphertext $C$ is transformed
into two parts $C_1, C_2$, and the $C_2$ is kept secret
from the PKG. We named this method ciphertext
diverge-merge scheme (CDMS). In our scheme,
only receivers can decrypt ciphertexts and the
amount of calculation computed on user’s
terminals is little.

To show that only receivers can decrypt
ciphertexts, we defined two attack models that
either the PKG and the service providing server or
the e-mail server collude with an adversary, and
proved that the proposed scheme is secure against passive attack in the models. Then, we made sure that our scheme needs less computation than existing schemes in private key extraction, encryption and decryption.

Future directions are extending the proposed scheme so that the scheme can be applied to IBEs other than Boneh and Franklin’s scheme and improving security so as to be secure against active attack.

7 REFERENCES

The International Journal of Digital Information and Wireless Communications aims to provide a forum for scientists, engineers, and practitioners to present their latest research results, ideas, developments and applications in the field of computer architectures, information technology, and mobile technologies. The IJDIWC publishes issues four (4) times a year and accepts three types of papers as follows:

1. **Research papers**: that are presenting and discussing the latest, and the most profound research results in the scope of IJDIWC. Papers should describe new contributions in the scope of IJDIWC and support claims of novelty with citations to the relevant literature.

2. **Technical papers**: that are establishing meaningful forum between practitioners and researchers with useful solutions in various fields of digital security and forensics. It includes all kinds of practical applications, which covers principles, projects, missions, techniques, tools, methods, processes etc.

3. **Review papers**: that are critically analyzing past and current research trends in the field.

Manuscripts submitted to IJDIWC should not be previously published or be under review by any other publication. Original unpublished manuscripts are solicited in the following areas including but not limited to:

- Information Technology Strategies
- Information Technology Infrastructure
- Information Technology Human Resources
- System Development and Implementation
- Digital Communications
- Technology Developments
- Technology Futures
- National Policies and Standards
- Antenna Systems and Design
- Channel Modeling and Propagation
- Coding for Wireless Systems
- Multiuser and Multiple Access Schemes
- Optical Wireless Communications
- Resource Allocation over Wireless Networks
- Security; Authentication and Cryptography for Wireless Networks
- Signal Processing Techniques and Tools
- Software and Cognitive Radio
- Wireless Traffic and Routing Ad-hoc Networks
- Wireless System Architectures and Applications