IJACSA
WHERE WISDOM SHARES
International Journal of Advanced Computer Science and Applications

Volume 6 Issue 3
March 2015

ISSN 2156-5570(Online)
ISSN 2158-107X(Print)

www.ijacsa.thesai.org

IJACSA

WHERE WISDOM SHARES

INTERNATIONAL JOURNAL OF
ADVANCED COMPUTER SCIENCE AND APPLICATIONS

THE SCIENCE AND INFORMATION ORGANIZATION

www.thesai.org | info@thesai.org

Editorial Preface

From the Desk of Managing Editor…

It may be difficult to imagine that almost half a century ago we used computers far less sophisticated than current home desktop computers to put a man on the moon. In that 50 year span, the field of computer science has exploded.

Computer science has opened new avenues for thought and experimentation. What began as a way to simplify the calculation process has given birth to technology once only imagined by the human mind. The ability to communicate and share ideas even though collaborators are half a world away and exploration of not just the stars above but the internal workings of the human genome are some of the ways that this field has moved at an exponential pace.

At the International Journal of Advanced Computer Science and Applications it is our mission to provide an outlet for quality research. We want to promote universal access and opportunities for the international scientific community to share and disseminate scientific and technical information.

We believe in spreading knowledge of computer science and its applications to all classes of audiences. That is why we deliver up-to-date, authoritative coverage and offer open access of all our articles. Our archives have served as a place to provoke philosophical, theoretical, and empirical ideas from some of the finest minds in the field.

We utilize the talents and experience of editor and reviewers working at Universities and Institutions from around the world. We would like to express our gratitude to all authors, whose research results have been published in our journal, as well as our referees for their in-depth evaluations. Our high standards are maintained through a double blind review process.

We hope that this edition of IJACSA inspires and entices you to submit your own contributions in upcoming issues. Thank you for sharing wisdom.

Thank you for Sharing Wisdom!

Managing Editor
IJACSA
Volume 6 Issue 3 March 2015
ISSN 2156-5570 (Online)
ISSN 2158-107X (Print)

Editorial Board

Reviewer Board Members

- **Abassi Ryma**
  Higher Institute of Communications Studies of Tunis , Iset'com
- **Abbas Karimi**
  Islamic Azad University Arak Branch
- **Abdelghni Lakehal**
  Université Abdelmalek Essaadi Faculté Polydisciplinaire de Larache Route de Rabat, Km 2 - Larache BP. 745 - Larache 92004. Maroc.
- **Abdel-Hameed A. Badawy**
  Arkansas Tech University
- **Abdur Rashid Khan**
  Gomal Unversity
- **Abeer Mohamed ELkorany**
  Faculty of computers and information, Cairo Univesity
- **ADEMOLA ADESINA**
  University of the Western Cape
- **Aderemi A. Atayero**
  Covenant University
- **Ahmed S.A AL-Jumaily**
  Ahlia University
- **Ahmed Boutejdar**
- **Ahmed Nabih Zaki Rashed**
  Menoufia University
- **Akbar Hossain**
- **Akram Belghith**
  University Of California, San Diego
- **Albert Alexander S**
  Kongu Engineering College
- **Alci-nia Zita Sampaio**
  Technical University of Lisbon
- **Alexandre Bouënard**
  Sensopia
- **Ali Ismail Awad**
  Luleå University of Technology
- **Amitava Biswas**
  Cisco Systems
- **Anand Nayyar**
  KCL Institute of Management and Technology, Jalandhar
- **Andi Wahju Rahardjo Emanuel**
  Maranatha Christian University
- **Andrews Samraj**
  Mahendra Engineering College
- **Anirban Sarkar**
  National Institute of Technology, Durgapur
- **Antonio Formisano**
- **Anuranjan misra**
  Bhagwant Institute of Technology, Ghaziabad, India
- **Appasami Govindasamy**
- **Arash Habibi Lashkari**
  University Technology Malaysia(UTM)
- **Aree Ali Mohammed**
  Directorate of IT/ University of Sulaimani
- **Aris Skander Skander**
  Constantine 1 University
- **Ashok Matani**
  Government College of Engg, Amravati
- **Ashraf Mohammed Iqbal**
  Dalhousie University and Capital Health
- **Ashraf Hamdy Owis**
  Cairo University
- **Asoke Nath**
  St. Xaviers College(Autonomous), 30 Park Street, Kolkata-700 016
- **Ayad Ghany Ismaeel**
  Department of Information Systems Engineering-Technical Engineering College-Erbil Polytechnic University, Erbil-Kurdistan Region- IRAQ
- **Ayman EL-SAYED**
  Computer Science and Eng. Dept., Faculty of Electronic Engineering, Menofia University
- **Babatunde Opeoluwa Akinkunmi**
  University of Ibadan
- **Badre Bossoufi**
  University of Liege
- **BASANT KUMAR VERMA**
  JNTU
- **Basil Hamed**
  Islamic University of Gaza
- **Basil M Hamed**
  Islamic University of Gaza
- **Bhanu Prasad Pinnamaneni**
  Rajalakshmi Engineering College; Matrix Vision GmbH
- **Bharti Waman Gawali**
  Department of Computer Science & information T

CONTENTS

Skew Detection/Correction and Local Minima/Maxima Techniques for Extracting a New Arabic Benchmark Database

Husam Ahmed Al Hamad
Department of Information Technology
Qassim University
Qassim, Saudi Arabia

***Abstract*—We propose a set of techniques for extracting a new standard benchmark database for Arabic handwritten scripts. Thresholding, filtering, and skew detection/correction techniques are developed as a pre-processing step of the database. Local minima and maxima using horizontal and vertical histogram are implemented for extracting the script elements of the database. Elements of the database contain pages, paragraphs, lines, and characters. The database divides into two major parts. The first part represents the original elements without modifications; the second part represents the elements after applying the proposed techniques. The final database has collected, extracted, validated, and saved. All techniques are tested for extracting and validating the elements. In this respect, ACDAR proposes a first issue of the Arabic benchmark databases. In addition, the paper confirms establishment a specialized research-oriented center refers to learning, teaching, and collaboration activities. This center is called "Arabic Center for Document Analysis and Recognition (ACDAR)" which is similar to other centers developed for other languages such as English.**

Keywords—ACDAR; Arabic benchmark database; Arabic scripts; document analysis; handwriting recognition; skew detection and correction

I. INTRODUCTION

Arabic language is spoken by hundreds of millions of people around the world. It profoundly influenced many cultures, including the Western culture, for many centuries. Although it is one of the most important languages in the world throughout its long history, it still lags behind many other languages as far as information technology resources and applications are concerned. As a result, the so-called "digital gap" is greater for Arabic language than other languages such as English, for instance.

Automatic recognition of handwritten words remains a challenging task even though the latest improvements of recognition techniques and systems are very promising. The term handwriting refers to some artificial graphical marks containing a message in a given human language [1]. The concept of handwriting has always existed, for the purpose of expanding people's memory and facilitating communication together [2] and much of the human culture may be attributed to the advent of handwriting. Because of the fact that only humans can perfectly understand and recognize the handwritings of others, one computationally challenging task resides in the attempt to imitate the human ability to read and recognize handwriting [3]. Consequently, automatic recognition of handwritten words remains a difficult task even though the latest improvements of recognition techniques and systems seem to be promising. For the purpose of automating Arabic scripts processing, numerous contributions have made in the area of handwritten script segmentation and recognition [4]. However, no outstanding results were reached so far, as OCR Arabic processing is still facing serious issues. One of the reasons is that Arabic language is considerably harder than Latin counterpart [5]. Therefore, in the area of automatic recognition of Arabic handwriting, many works have still to be done. One of the most important requirements for the development and comparison of recognition systems is a large database together with ground truth information. Compared to Latin scripts where handwritten words and numbers have publicly available for a long time (e.g. CEDAR [6], NIST[1]) the situation for Arabic is quite different. Others implement large databases that are not available to the public [7], or unreliable databases that concern only one Arab country (e.g. IFN/ENIT [8]).

Although many research efforts have done, so far in the recognition of handwritten Arabic script [4] until now they have not reached satisfactory results for the following reasons [2].

- Arabic words are overlapped and written always cursively, i.e., more than one character can be written connected to each other.
- Arabic writing uses many types of external objects, such as 'dots', 'Hamza', 'Madd', and diacritic objects, these external objects make the task of line separation and segmentation scripts more difficult.
- An Arabic character can have more than one shape according to its position in the word, i.e., initial, middle, final, or as a standalone character.
- Arabic writing uses many ligatures, especially in handwritten text.
- Other characters have very similar contours and are difficult to segment and to recognize especially when non-characters and external objects are present in the

[1] NIST database, http://www.nist.gov/srd/

scanned image.

II. HISTORICAL BACKGROUND

Earlier surveys discussed recognition and segmentation of both handwriting and machine-print, with much emphasis on machine-print. Unfortunately, only a small and unreliable database is available for Arabic Language today. However, in 1980, Nouh et al. suggested a standard Arabic character set to facilitate computer processing [9]. Standard and reliable databases were developed many years ago for the recognition of handwriting in Latin scripts. Among these databases, the CEDAR database (Center of Excellence for Document Analysis and Recognition) was released in 1993 [6]. It contains images of approximately 50,000 alphanumeric characters, 5,000 city names, 5,000 state names, and 10,000 ZIP codes. Each image was scanned from mail in a working post office at 300 pixels per inch in 8-bit grayscale on a high quality flatbed digitizer. The data were unconstrained for the writer, style and technique of preparation. These characteristics help overcome the limitations of earlier databases that contained only isolated characters or were prepared in a laboratory setting under prescribed circumstances. In addition, the database is divided into explicit training and testing sets to facilitate the sharing of results among researchers as well as performance comparisons.

In 1999 Al ISRA Arabic database [10] collected from 500 students, it contains words, digits, signatures, which is has limitation because it does not contain paragraphs. Another database lunched in 2002 is IFN/ENIT [4, 11], it was developed at the Institute of Communications Technology (IFN) at Technical University Braunschweig in Germany and the Ecole Nationale d'Ingenieurs de Tunis (ENIT) in Tunisia. It consists of 26,459 images of the 937 cities names and towns in Tunisia, written by 411 different persons filled forms with about 26400 names containing more than 21,0000 characters. For each name some information are coded such as the sequence of character shapes, some style information, and the baseline are coded. It is used for recognition of data entry, mail sorting, and other recognition tasks. The images are partitioned into four sets so that researchers can use and discuss training and testing data in this context. The database has certainly many advantages and some of its drawbacks is the fact that it is written for Tunisia only and therefore contains only Tunisian cities and names and does not cover other Arab countries. It also lacks reliable training and testing sets. As a result, it is not widespread among researchers.

One of the efforts that addressed the handwriting recognition problem is in writing personal checks. One such system, developed a decade ago, is AHDB (Arabic Handwritten DataBase), a database containing samples from 100 different writers, including words used for numbers [12]. In 2003 AI-Ohali et al., developed CENPARMI images databases from 3,000 checks and implemented at the Center for Pattern Recognition and Machine Intelligence provided by a banking corporation [13].

These databases contain numeric amounts written in words, sub-words, Indian digits, and numeric amounts written with Indian digits. Notably, Indian digits are the numeric digits normally used in Arabic writing, as opposed to “Arabic numerals” ordinarily used in Latin script. The Indian digits database contains 15,175 samples, the legal and courtesy databases 2,499 samples, and the sub-words database contains 29,498 samples.

In 2009 ADAB database (Arabic DAtaBase) with Arabic online handwritten words has used by Haikal, *et al* [14] at the first time, the database was developed for Arabic online handwritten scripts in a cooperation between the Institute for Communications Technology (IfN) and the Ecole Nationale d’Ing`enieurs de Sfax (ENIS). The database written by more than 130 persons, it consists of 15158 Arabic handwritten words, 937 Tunisian town/village names. The database contains in additional special tools for the collection of the data and verification of the ground truth. These tools give the possibilities to record the online written data, to save some writer information, to select the lexicon for the collection, and re-write and correct wrong written text.

Although the recognition accuracy for separated handwritten numerals and characters has improved significantly in recent years, the final frontier remains the accurate recognition of handwritten Arabic scripts. The pursuit of more accurate recognition rates continues to encourage researchers in the field. It must also be mentioned that along with the challenging nature of the handwritten word recognition problem, immense potential lies in the commercial sector to make these systems available. So, an important bulk of work is required to undertake a serious research and meet its multiple challenges.

III. BENCHMARK DATABASE

One of the most important components in ACDAR center is the benchmark database; often recognition algorithms have tested using one type of database, especially in the case of off-line handwriting recognition. ACDAR is concerned with both off-line and online handwriting recognition. It proposes a common benchmark database of Arabic handwritten scripts, which is essential for research on handwritten Arabic word recognition. The first issue of this database has hosted in the center.

ACDAR began to work with the off-line Arabic handwriting recognition, which is may divided into segmentation-based and holistic ones. In general, the former approach uses a strategy based on the recognition of individual characters or patterns whereas non-segmentation based deals with the recognition of the word image as a whole [15]. In the online case, the handwriting has captured and stored in digital form *via* different means. Usually, a special pen has used in conjunction with an electronic surface. As the pen moves across the surface or paper, the two-dimensional coordinates of successive points have represented as a function of time and have stored in order [1]. It is generally, information is not easy to recover from handwritten words written on a non-digital medium such as accepted that the online technique of recognizing handwriting has so far achieved better results than off-line. This may be attributed to the fact that more information may be captured in the online case such as the direction, speed and the order of strokes of the handwriting. At the end, ACDAR's database will be made freely available to researchers.

A. Data collection

ACDAR started recognizing the paragraphs, lines, words, and characters. The handwriting papers have wrote by 113 distinct writers and scanned in a RGB-scale. The writers are variant in age, education, background, genders, and countries; Figure 1 shows a snapshot form that contains the instruction and personal details of the writers, Table I shows the statistical data that collected from all writers. Figure 2 illustrates the comparisons of database contains.

As a result, two paragraphs contain all shapes of Arabic characters have wrote by those writers. Figure 3 shows the original two paragraphs have requested to write by the persons, Figure 4 displays in blue the position of the different characters shapes, the second paragraph has required for collecting more samples.

Fig. 1. Form of the personal details of each writers

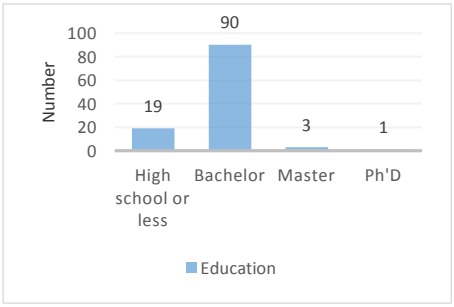
TABLE I. STATISTICAL DATA OF 113 WRITERS

Education	High school or less	Bachelor	Master	PhD
Number of writers	19	90	3	1

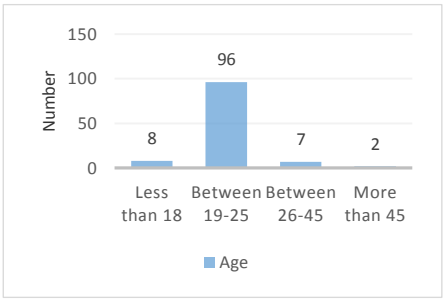
Age	Less than 18	Between 19-25	Between 26-45	More than 45
Number of writers	8	96	7	2

Gender	Male	Female
Number of writers	72	41

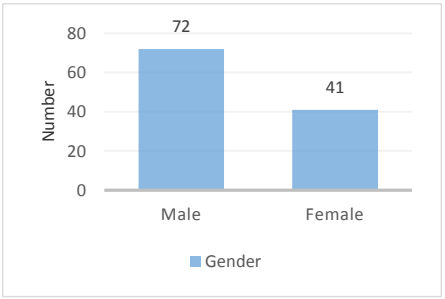
Country	Saudi Arabia	Jordan	Algeria	Syria	Egypt	Yemen
Number of writers	85	20	2	3	1	2



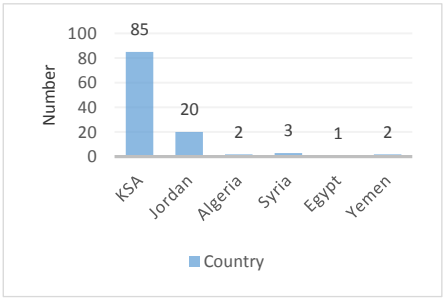
a) education



b) age



c) gender



d) country

Fig. 2. Comparisons of the statistical, a) education, b) age, c) gender, and d) country

Lines, words and characters have also extracted and saved in the database; verification phase has also investigated before the final adoption of the samples for quality purposes. In summary, each writer has wrote 358 words (first paragraph contains 162 words, second paragraph contains 196 words), both of them are 1,916 characters, on average each writer has wrote 30 lines. In total, all writers have wrote 226 paragraphs, 3,390 lines, 40,454 words, and 216,508 characters. Number of 57 writers are identified what they wrote as a training set, also 56 are writers identified what they wrote as a testing set. Table II illustrates the numbers of images have collected before extraction and validation processes.

Fig. 3. Two paragraphs cover all shapes of Arabic characters

Fig. 4. Position of all Arabic characters shapes covered by only the first paragraph

TABLE II. NUMBERS OF COLLECTED IMAGES BEFORE EXTRACTION AND VALIDATION PROCESSES

Details / writers	Each writer	Training set by 57 writers	Testing set by 56 writers	Total (training and testing sets) 113 writers
Paragraphs	2	114	112	226
Lines	30 as average	1,710	1,680	3,390
Words	358	20,406	20,048	40,454
Characters	1,916	109,212	107,296	216,508

The images have scanned in 200, 300 dpi resolution in RGB-scale images [2, 16, 17, 18]. Two version of the scanned documents have saved in the database, before and after preprocessing. Paragraphs, lines, words, and characters have extracted and saved as well.

The key steps of the techniques that were developed in this research is shown in Figure 5, the Figure shows briefly how we extracted all paragraph, lines, words, and characters. The first step is scanning the original documents with 200 and 300 dpi in RGB-scale, next step is preprocessing the scanned images, skew detection / correction, thresholding, and remove the noise using filtering are investigated in this stage. Next, start extracting process of the database; this stage includes developing a set of techniques to get the best extraction results of lines, words and characters. Finally, the last step is validating step, all extracted elements underwent to the evaluation process, if the element successfully passed this stage, then it will save into the benchmark database, otherwise it will discard. As mentioned before, this research aims to build the first issue of ACDAR database and test the proposed algorithms have developed in this research.

B. Pre-processing

Many techniques have developed to perform further processing to allow superior recognition. Thresholding and filtering which they aim to eliminate and remove any noise or any small ascenders. Skew detection and correction technique that aims to adjust slopes of the paragraphs and lines. Next sub-sections are explain in details the parts of the pre-processing.

1) Thresholding and Filtering

The first step of preprocessing is thresholding (binary format); it uses as prior to further processing. Thresholding involves the conversion of a grey-scale image (0–255) into a binary image (0–1). This format will be easier to manipulate an image without levels of color in some researches, in additional the processing will be faster, less computationally expensive and will allow for more compact storage. The goal of using the thersholding is to determine the segmentation points of the lines, words, and the characters. Determine the segmentation points from the grey-scale image will be easier than color the image; the same points have extracted were applied on the RGB-scale. There are of course many of the defects such as loosing features from image. However, since the goal of this stage is only to determine the segmentation points, the effect will be the lowest grades possible. *rgb2gray* function was used to converts RGB images into grayscale by eliminating the hue and saturation information while retaining the illumination. The definition *rgb2gray* is shown in the following equations. The *im2bw* function was also used to convert this grayscale image to binary format (matrix). The output binary image BW has values of 0 as a foreground pixel (black) for all pixels in the input image and 1 as a background pixel (white) for all other pixels. All images were converted using the previous technique so that only binary images remained and could be used for further processing.

$$g = w_r I_r + w_g I_g + w_b I_b \quad (1)$$

$$\text{s.t. } w_r + w_g + w_b = 1, \quad (2)$$

$$w_r \geq 0, w_g \geq 0, w_b \geq 0, \quad (3)$$

where, g is a constraint linear combination of R, G and B channels of input color image I,

I_r, I_g, and I_b are the inputs,
w_r, w_g, and w_b : weights sum to 1, and they are non-negative numbers.

Next, elimination of the elements noise; the goal of this technique is to remove the noise as well as small foreground objects that were not part of the writing. Once the component of word image as matrix were identified, it was possible to perform various useful operations. *imfilter* function has used, it performs multidimensional filtering according to the specified options like *fspecial* function to create 2-D special filters that used ‘disk’ function to returns a circular averaging filter 'pillbox' within the square matrix of side (2 * radius + 1). Gaussians function [19] is applied, it was used at the lowest degree possible in order to not lose the features of the scripts

as much as possible. The function aims to make the word image more smoothly, and to eliminate any small ascenders or descanters noise between the lines. The following equation shows the one per direction using Gaussians function.

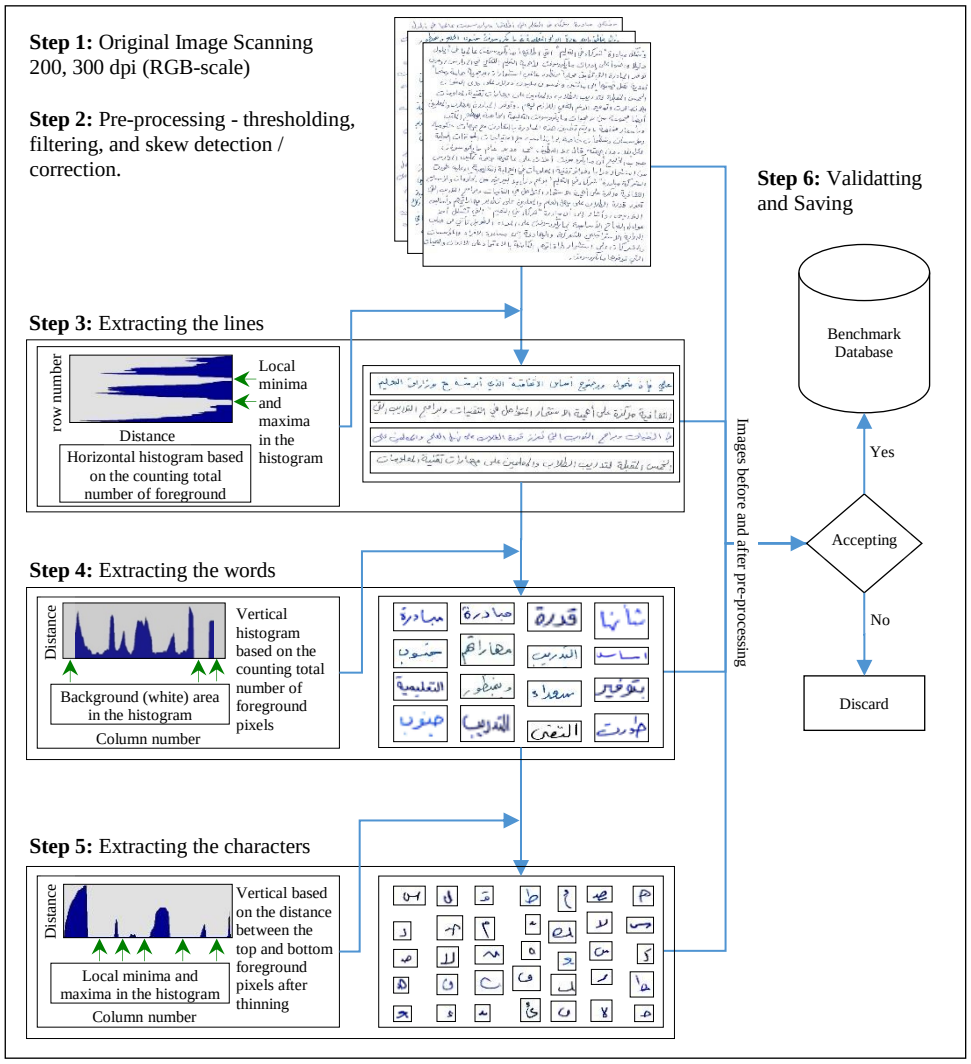


Fig. 5. Steps of extracting and validating the benchmark database

$$g(x,y) = \frac{1}{2\pi\sigma^2} \quad . \quad e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (4)$$

where, σ is the standard deviation of the Gaussian distribution,

x is the distance from the origin in the horizontal axis,

y is the distance from the origin in the vertical axis.

2) Skew Detection and Correction

Before extracting the lines, words, and characters from the documents images, skew of the paragraphs should be detect and then correct, the technique uses projections of an image matrix along specified directions, Hough transform [20] algorithm is applied to detect and correct the slopes. Hough Transform is the linear transform for detecting straight lines, the straight line is described as $y = mx + b$ where the parameter m is the slope of the line, and b is the intercept (*y-intercept*). Before start to apply the Hough transform algorithm, the document image should be prepared. So, a set of steps were used, at the beginning Threshold of the image to binary was applied. Next, in order to obtain a clear base line for all lines in the page, the punctuation marks (dots) and small stroke have removed from the image. Then, dilate image is also applied to close the internal gaps between the characters and words as well. Closing operation was performed upon the horizontal line element and merging the words of the lines. The text lines now look likes rectangles, to apply the Hough transform one-step is remained, this step is thinning the image includes all horizontal rectangles. To find the skew of the image, the mean and standard deviation of slopes were calculated, any bad data conceders far away from the standard deviation was removed, then the average of the good slopes was calculated, therefore the skew can be calculated by using the angle of the slope. The skew correction has applied using the negative of this angle. The below equations shows the Hough transform technique, the line

equation can be written as shown in equation (5), rearranged the equation shown in equation (6), and equation (7) shows formula of an point on the image with coordinates. Figure 6 shows the steps of skew detection and correction have developed in this research; Figure 7 shows samples of skew detection and correction for one paragraph and one line.

$$y = \left(-\frac{\cos\theta}{\sin\theta}\right)x + \left(\frac{r}{\sin\theta}\right) \quad (5)$$

$$r = x\cos\theta + y\sin\theta \quad (6)$$

$$r(\theta) = x_0\cos\theta + y_0\sin\theta \quad (7)$$

where, r distance between the line and the origin, is determined by θ,

θ is the angle of the vector from the origin to this closest point.

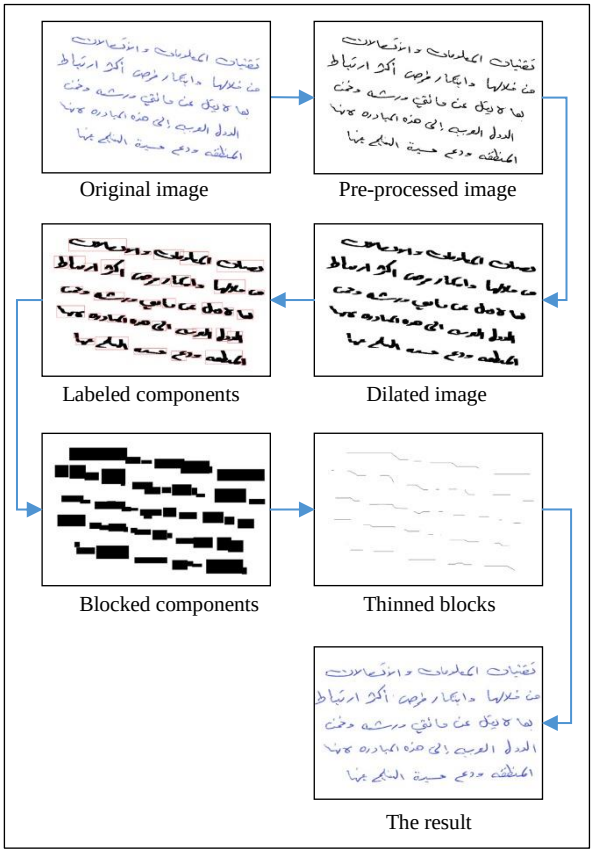


Fig. 6. Steps of skew detection and correction

C. Extracting the database

Local minima and maxima of horizontal and vertical histograms have used for determining the segmentation points SPs for extracting the lines, words and characters. The concept of using the horizontal histogram is for extracting the line image. Horizontal histogram is formed by counting the total numbers of foreground pixels (black color) for each row from left to right in the paragraph image; the segmentation points have located based on the white color (background pixel) or the distance between two successive local maxima and one local minima with almost no foreground pixels. Figure 8 illustrates technique of extracting the lines images based on horizontal histogram.

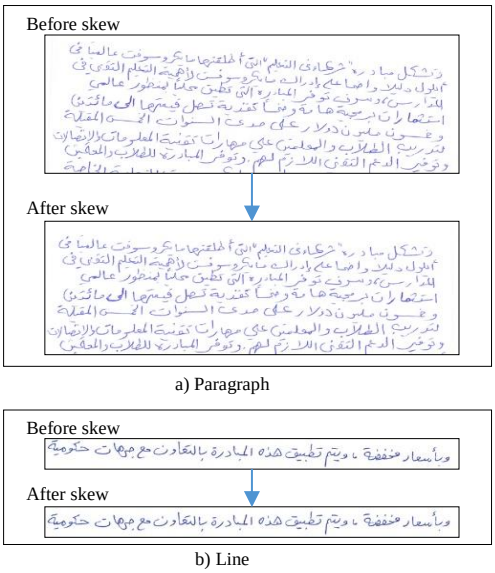


Fig. 7. Sample of skew detection and correction, a) a paragraph b) a line

Using the same technique, but now by applying the vertical histogram to extract the word images. Vertical histogram has formed by counting the total numbers of foreground pixels (black color) for each column from top to bottom in the line image. The segmentation points have located based on the white color (background pixel) or the distance between two successive local maxima and one local minima with almost no foreground pixels. Figure 9 illustrates technique of extracting the words images based on vertical histogram.

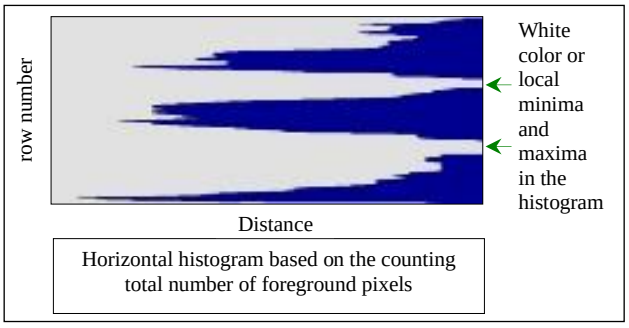


Fig. 8. Extracting the lines images based on horizontal histogram

Likewise, extracting the characters images technique uses also the vertical histogram, which has calculated based on the distance between the top and bottom of foreground pixels for the word image after thinning. Extracting of the characters from the words is required to remove the punctuation marks (dots). The dots here consider a major obstacle to identify the correct segmentation points of the characters. After determining the segmentation points, the dots will recover. Figure 10 illustrates technique of extracting the word images based on vertical histogram before thinning and Figure 11 after thinning.

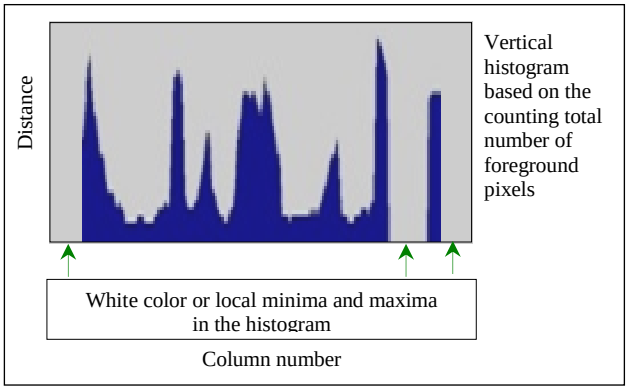


Fig. 9. Extracting the words images based on horizontal histogram before thinning

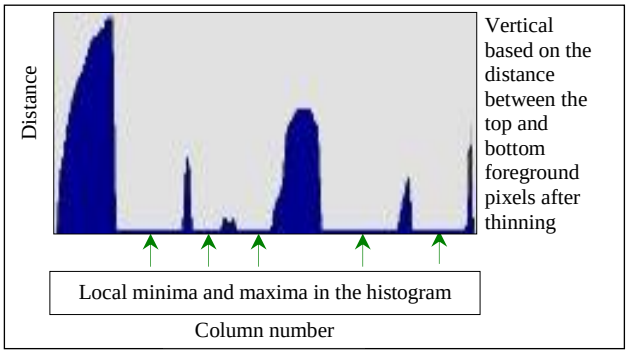


Fig. 10. Extracting the word images based on horizontal histogram after thinning

The following equations show how the histogram has calculated based on the total number of the foreground pixels.

$$pr(r_k) = \frac{n_k}{n} \quad (8)$$

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n} \quad (9)$$

where, K=0, 1, …., L-1,
N is total number of pixels in the image,
L is total number of possible grey levels in the image.

IV. Experimental Results

As a result of all the previous steps, in addition to the final stage of verification processes, the first issue of ACDAR database is now available. ACDAR database contains 208 pages, 208 paragraphs, 2,969 lines, 32,890 words, and 158,872 characters, the database is divided into two sets one for training and the second for testing. The details of the first issue of ACDAR database after extraction and validation process is shown in Table III. Table IV summarizes a comparison between the results of some databases use the Arabic handwritten scripts. Diversified samples from the ACDAR database have published in ACDAR's website under this link http://www.acdar.org/DBsamples.php.

Figure 12 displays samples of handwritten paragraph that written by one person with its printed text, Figures 13 to 16 display samples of complete free handwritten paragraph. More samples for characters, words, and paragraphs see http://www.acdar.org/ DBsamples.php.

TABLE III. Final Database after Extracting and Validating Processes

Details / writers	Each writer	Training set by 51 writers	Testing set by 53 writers	Total (training and testing sets) 104 writers	Percentage from the original
Paragraphs	2	102	106	208	92.0%
Lines	Average 30	1,467	1,502	2,969	87.6%
Words	358	16,214	16,676	32,890	81.3%
Characters	1,916	78,584	80,288	158,872	73.4%

TABLE IV. Comparison Between Some of Handwritten Databases Used Arabic Letters

Database	Details	Year	Writers
Al-Isra [10]	▪ 500 sentences ▪ 37,000 words ▪ 10,000 digits ▪ 2,500 signatures	1999	500
AHDB [12]	▪ 10,000 words for check processing	2002	100
IFN/ENIT [8]	▪ 26,459 Tunisian city names	2002	411
Khedher and Abandah [21]	▪ 48 pages of text	2002	48
IFHCDB [22]	▪ 52,380 characters ▪ 17,740 numerals	2006	–
ADBase / MADBase [22]	▪ 70,000 digits	2007	700
Alamri et al. [24]	▪ 11,375 words ▪ 21,426 characters ▪ 46,800 digits strings ▪ 13,439 numerical ▪ 1,640 special symbols	2008	328
On/Off LMCA [25]	▪ 500 words ▪ 1,00,000 characters ▪ 30,000 digits	2008	55
Al Hamad et al. [2]	▪ 20 Pages ▪ 40 paragraphs ▪ 500 words ▪ 620 characters	2010	10
ACDAR – First issue	**▪ 208 Paragraphs/Pages ▪ 2,969 Lines ▪ 32,890 words ▪ 158,872 characters**	**2014**	**113**

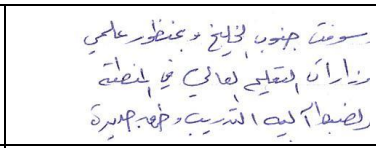
handwritten	
printed	سوفت جنوب الخليج وبمنظور علمي وزارات التعليم العالي في المنطقة لضبط آلية التدريب وطرق جديدة

Fig. 11. Sample of part free handwritten paragraph

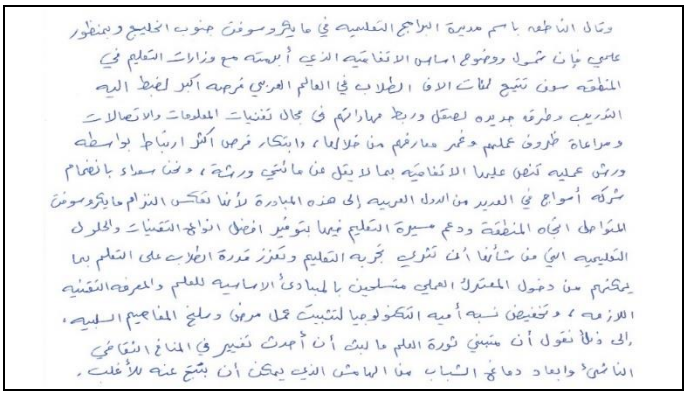


Fig. 12. Sample (a) of complete free handwritten paragraph 1

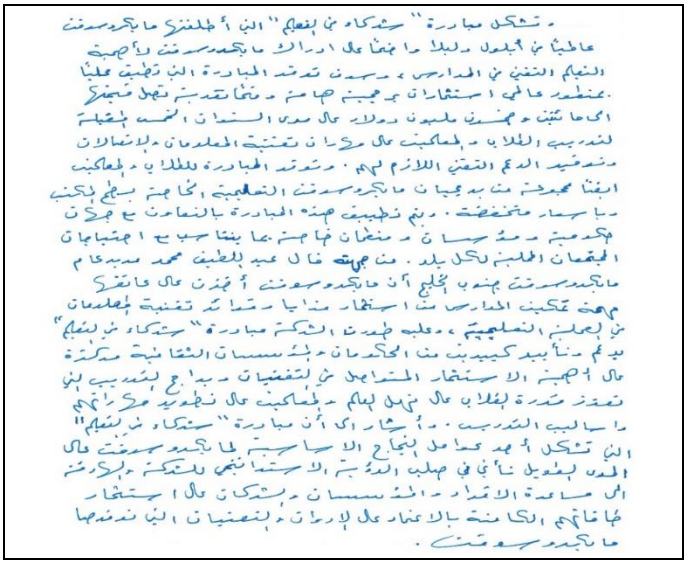


Fig. 13. Sample (a) of complete free handwritten paragraph 2

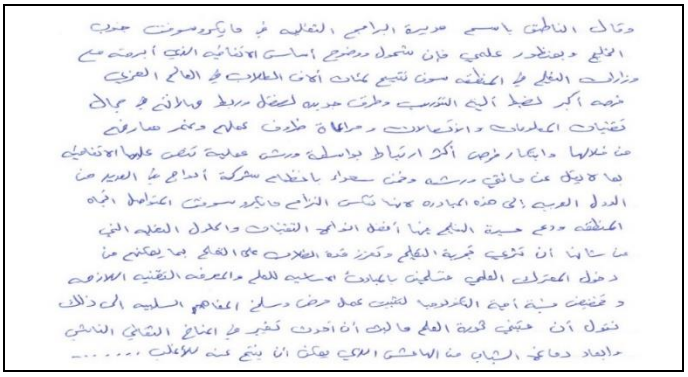


Fig. 14. Sample (b) of complete free handwritten paragraph 1

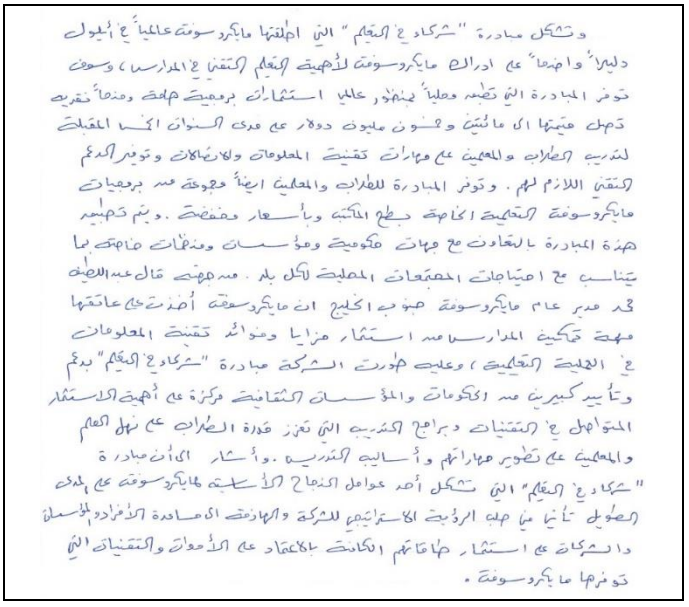


Fig. 15. Sample (b) of complete free handwritten paragraph 2

Figures 17 and 18 show samples of free handwritten lines extracted from the paragraphs.

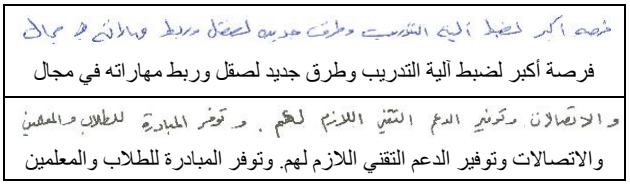


Fig. 16. Samples of ACDAR free handwritten lines

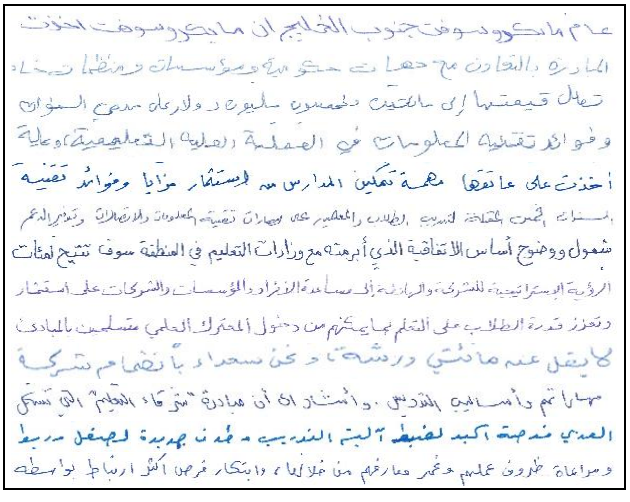


Fig. 17. Samples of ACDAR free handwritten lines written by different persons

In addition, the first issue of the ACDAR database contains different samples of Arabic handwritten words; Figure 19 shows samples of handwritten words with their printed text, Figure 20 shows samples of free handwritten extracted from the lines image.

A sample of the characters that extracted from the words image and wrote by one person is shown in Figure 21; Figure

22 shows samples of free handwritten characters wrote by many writers.

Handwritten	Printed	Handwritten	Printed
الذي	الذي	وقال	وقال
ودعم	ودعم	ومراعاة	ومراعاة
أحد	أحد	التعليم	التعليم
الدول	الدول	امواج	امواج
الطلاب	الطلاب	أساس	أساس
مساعدة	مساعدة	وزارات	وزارات
ارتباط	ارتباط	وابتكار	وابتكار
باسم	باسم	وغمر	وغمر

Fig. 18. Samples of ACDAR free handwritten words

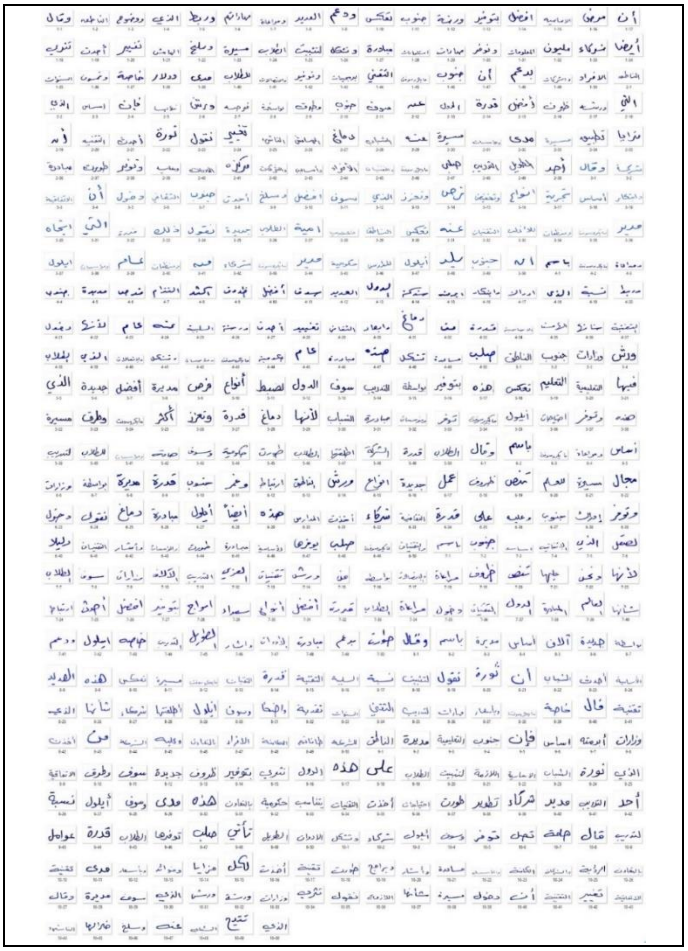


Fig. 19. Samples of ACDAR free handwritten characters written by different persons

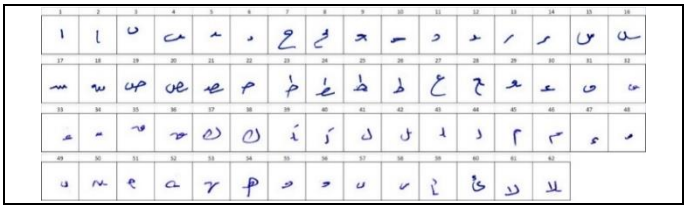


Fig. 20. Sample of ACDAR character that wrote by one person

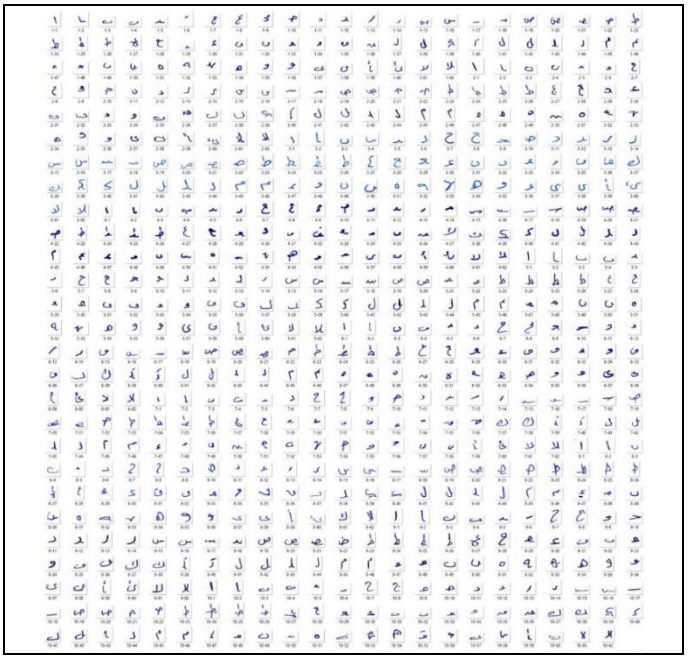


Fig. 21. Samples of ACDAR free handwritten characters written by different persons

V. ACDAR – A NEW CENTER

As any other center in this area, ACDAR [26] contains a set of internal sections describing the main objectives of the center and its contents, these sections cover many activities in the area of document analysis and recognition such as teach courses, research, publications, resources, people, contact details. While analysis of documents and handwriting recognition continues to be our primary interest, we propose research and software development projects involving diverse digital document types.

ACDAR is dedicated to re-build and re-structure a reliable and standard a benchmark database and set of integrated tools for handwritten Arabic scripts within, it is newly established a website (http://www.acdar.org). The website includes details about the center, and a sample of the first issue of the benchmark database. In the conceptual framework of ACDAR, many functions would give ACDAR its identity, mission, and direction. These centered on the benchmark database, research, training, and collaboration with the community. More details about ACDAR center see Al Hamad *et al* [26].

VI. CONCLUSION

The paper presents new techniques for extracting the first issue of a new benchmark database, it has written by 113 distinct writers with different ages, cultures, and genders. Two paragraphs cover all shapes of Arabic characters have scanned with different resolution; the final database contains 208 pages, 208 paragraphs, 2,969 lines, 32,890 words, and 158,872 characters. Half of the database assigns as training set; another part assigns as testing set. For extracting and validating the proposed database, the research has developed and tested a set of new techniques. An example of these techniques are pre-processing of the images such as

thresholding, filtering, local minima and maxima of vertical and horizontal histogram for the segmentation, in addition, developing skew detection / correction technique, etc. The techniques have examined and tested through several experiments in order to use them later for creating a comprehensive database that we seek to cover all Arab countries. The paper also displays a comprehensive details of forming a new center for analysis and recognition Arabic handwritten scripts, the center calls "ACDAR". Functions and activities of the center have identified and explained in detail.

REFERENCES

[1] Plamondon, R., S.N. Srihari, "On-line and Off-line Handwriting Recognition: A Comprehensive Survey," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, 2000, pp. 63–84.

[2] Al Hamad, H.A., R. Abu Zitar, "Development of an Efficient Neural-based Segmentation Technique for Arabic Handwriting Recognition," Pattern Recognition, vol. 43(8), 2010, pp. 2773-2798.

[3] Blumenstein M., "Intelligent Techniques for Handwriting Recognition. School of Information Technology," PhD Dissertation, Griffith University-Gold Coast Campus, Australia, 2000.

[4] Lorigo, L., V. Govindaraju, "Off-line Arabic Handwriting Recognition: A Survey," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28(5), 2006, pp. 712–724.

[5] Hamid, A., R. Haraty, "A Neuro-Heuristic Approach for Segmenting Handwritten Arabic Text," ACS/IEEE International Conference on Computer Systems and Applications, AICCSA, vol.1, 2001, pp. 1–10.

[6] Hull, J., "Database for Handwritten Text Recognition Research, Center of Excellence for Document Analysis and Recognition (CEDAR)," Department of Computer Science, State University of New York at Buffalo, Buffalo, New York, http://www.cedar.buffalo.edu/ Databases/CDROM1, 1993.

[7] Kharma, N., M. Ahmed, R. Ward, "A New Comprehensive Database of Hand-written Arabic Words," Numbers and Signatures used for OCR Testing. IEEE Canadian Conference on Electrical and Computer Engineering, 1999, pp. 766-768.

[8] Pechwitz, M. et al., "IFN/ENIT – Database of Handwritten Arabic Words, Proc. of CIFED," 2002, pp. 129-136.

[9] Nouh, A., A. Sultan, R. Tolba, "An Approach for Arabic Characters Recognition," J. Eng. Sci, vol. 6, 1980, pp. 185–191.

[10] N. Kharma, M. Ahmed, R. Ward, "A new comprehensive database of handwritten Arabic words," numbers, and signatures used for OCR testing, Canadian Conference on Electrical and Computer Engineering, 1999, pp. 766–768.

[11] Pechwitz Mario, et al, "IFN/ENIT – Database of Handwritten Arabic Words," Institute for Communications Technology (IFN), Technical University Braunschweig, Germany, Ecole Nationale d'Ingénieur de Tunis (ENIT), BP 37 le Belvédère 1002, Tunis. IFN/ENIT, 2002, http://www.ifnenit.com/.

[12] Alma'adeed, S., D. Elliman, C. A. Higgins, "A Database for Arabic handwritten Text Recognition Research," Eighth International Workshop on Frontiers in Handwriting Recognition, 2002, pp. 485-489.

[13] AI-Ohali, Y. M. Cheriet, and C. Suen, "Databases for Recognition of Handwritten Arabic Cheques," Pattern Recognition, vol. 36, 2003, pp. 111-121.

[14] Haikal El Abed, et al, "Online Arabic Handwriting Recognition Competition," 10th International Conference on Document Analysis and Recognition ICDAR, 2009, DOI 10.1109/ICDAR.2009.284.

[15] Fan, X., B. Verma, "Segmentation vs. Non-Segmentation Based Neural Techniques for Cursive Word Recognition," An Experimental Analysis International Journal of Computational Intelligence and Applications, vol. 2(4), 2002, pp. 377–384.

[16] Al Hamad, H.A., "Over-segmentation of handwriting Arabic scripts using an efficient heuristic technique," IEEE International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR), 2012, pp.180-185.

[17] Al Hamad, H.A., "Neural-Based Segmentation Technique for Arabic Handwriting Scripts," WSCG 2013, 21st International Conference on Computer Graphics, Visualization and Computer Vision, indexed by Thomson Reuters/ISI-WoS, Czech, June, 2013.

[18] Al Hamad, H.A., "Use an Efficient Neural Network to Improve the Arabic Handwriting Recognition," IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 2013.

[19] Nixon M. S. and Alberto S. Aguado, "Feature Extraction and Image Processing. Academic Press," 2008, pp. 88.

[20] Richard O. Duda and Peter E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," ACM, vol. 15(1), 1972, pp. 11-15, doi:10.1145/36:1237.361242.

[21] Khedher, M., Abandah, G., "Arabic character recognition using approximate stroke sequence," Arabic Language Resources and Evaluation - Status and Prospects Workshop, 3rd International Conference on Language Resources and Evaluation (LREC'02), 2002.

[22] Mozaffari S., Faez k., Faradji F. Ziaratban M, Golzan S. M., "A comprehensive isolated Farsi/Arabic character database for handwritten OCR research," In Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition (IWFHR), 2006, pp. 385–389.

[23] El-Sherif E., Abdelazeem S., "A two-stage system for Arabic handwritten digit recognition tested on a new large database," In Proceedings of the International Conference on Artificial Intelligence and Pattern Recognition (AIPR'07), 2007, 237–242.

[24] Alamri H., He C. L., Sue C. Y., "A new approach for segmentation and recognition of Arabic handwritten touching numeral pairs," Proceedings of the International Conference Computer Analysis of Images and Patterns (CAIP). Lecture Notes in Computer Science, vol. 5702, Springer, 2009, pp. 165–172.

[25] Kherallah Monji , Elbaati A., El Abed H., Alimi A. M., "The on/off (LMCA) dual Arabic handwriting database," 11th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2008.

[26] Al Hamad, H.A., Hamdi-Cherif A., "The Arabic Center for Document Analysis and Recognition (ACDAR) - Structure and Perspective," European Conference of COMPUTER SCIENCE (ECCS '12), 2012.

Android Application to Assess Smartphone Accelerometers and Bluetooth for Real-Time Control

M.A. Nugent
Faculty of Engineering and Computing
Dublin City University, Dublin 9, Ireland

Dr. Harold Esmonde
Faculty of Engineering and Computing
Dublin City University, Dublin 9, Ireland

***Abstract*—Modern smart phones have evolved into sophisticated embedded systems, incorporating hardware and software features that make the devices potentially useful for real-time control operations. An object-oriented Android application was developed to quantify the performance of the smartphone's on-board linear accelerometers and bluetooth wireless module with a view to potentially transmitting accelerometer data wirelessly between bluetooth-enabled devices. A portable bluetooth library was developed which runs the bluetooth functionality of the application as an independent background service. The performance of bluetooth was tested by pinging data between 2 smartphones, measuring round-trip-time and round-trip-time variation (jitter) against variations in data size, transmission distance and sources of interference. The accelerometers were tested for sampling frequency and sampling frequency jitter.**

Keywords—Android; Bluetooth; control; real-time; sensors; smartphone

I. INTRODUCTION

Smartphones evolved from the PDAs of the late 1990s. PDAs were handheld computers essentially used for organising information. They were equipped with small keyboards that the user could utilise to input information. IBM Simon was the first PDA with mobile phone functionality and can be considered the first smartphone. In 2007 Apple Inc. introduced the iPhone which incorporated a large multi-touch screen for direct finger touch input as its main method of interaction.

Mass-produced ARM based microprocessor technology delivers high speed multi-processing on an inexpensive, battery powered platform, that only a decade ago, industrial computers would have been envious of. Smartphones have a myriad of onboard sensors, such as motion sensors including accelerometers and gyroscopes, environmental sensors that can measure pressure, light, temperature and humidity as well as position sensors such as orientation sensors, magnetometers and GPS locators. The original purpose of the smartphone was for communication and smartphones have expanded their capabilities here also including bluetooth and infrared.

Smartphones comprise 2 operating systems, a low-level operating system that handles the drivers for the hardware and a higher level user-interfacing operating system. The most common operating system installed worldwide is Google Inc.'s Android operating system which is built on top of a Linux kernel.

Android was first established in 2003 with the aim of developing a more user oriented operating system than Symbian and Microsoft. The main advantages of Android over its rivals are its flexibility and upgradability. Android has grown in popularity among consumers and developers alike to the point where an industry survey [1] in 2013 shows that 71% of all mobile development is for the Android operating system.

Since its inception there have been many evolutions of the Android operating system dashboard, from the original 'Froyo' through to 'KitKat' shipped with new smartphones. Table 1 displays the various distributions of Android dashboards.

TABLE I. ANDROID DASHBOARDS AND DISTRIBUTION LEVELS

Version	Codename	API	Distribution
Froyo	2.2	8	0.7%
GingerBread	2.3.3 – 2.3.7	10	13.6%
Ice Cream Sandwich	4.0.3 – 4.0.4	15	10.6%
JellyBean	4.1.x	16	26.5%
JellyBean	4.2.x	17	19.8%
JellyBean	4.3.	18	7.9%
KitKat	4.4	19	20.9%

Developing an application that is compatible with API 10 and higher will guarantee coverage of 99.3% of the Android market, but older APIs are not compatible with newer android features. Some features are only available with more recent APIs due to continuous developments by Google. Developers need to be aware of the features available with each version and the size of the market associated with that version. The bluetooth application on which this paper is based is compatible with all dashboards from Froyo to KitKat.

The Android accelerometers are primarily intended for screen orientation and game play. Android [2] describes the accelerometer sampling periods in terms of data delays in sending sensor readings to the application, ranging from 200,000 microseconds for the 'Normal' sampling rate to 0 microsecond delay for the 'Fastest' sampling rate. The delay is only a suggested delay and the Android system and other applications can change it.

Bluetooth is a wireless communication protocol invented by Ericsson in 1994 as a wireless replacement for serial port communications between mobile phones and headsets [3, 4]. Management of the specification passed to the Bluetooth Special Interest Group (BSIG) in 1998 and Bluetooth 1.0 was released in 1999 with a data rate of 721 kbit/s. Bluetooth 2.0 Enhanced Data Rate (EDR) was adopted in 2004, providing a data rate of 1 Mbit/s without EDR and 3 Mbit/s with EDR, and coinciding with the landmark of 3 million product shipments per week mark. In 2009 Bluetooth 3.0 High Speed (HS) was adopted with a data rate of up to 24 Mbit/s and Bluetooth 4.0 Low Energy was adopted in 2012 when annual product shipments exceeded 2 billion. Current specification development is in the area of IP connectivity preparing bluetooth for the Internet of Things revolution.

Bluetooth is a low energy, short-range, short wavelength radio transmission protocol operating within the unlicensed ISM radio frequency band from 2.4 – 2.485 GHz. A bluetooth radio can have a range from 1 meter up to 100 metres, depending on the class of device with smartphones typically ranging up to 10 metres. Once connected, a small network called a piconet is dynamically created which allows a master device to connect with up to 7 slave devices [5]. Each device can be connected to multiple piconets simultaneously allowing for complex, wide-ranging connectivity. One of the main advantages of bluetooth networks is their ease of set up. Two devices can connect with the push of a button with little configuration required from the user.

Research [6, 7] has been carried out into the performance of bluetooth over varying distances, data sizes and sources of interference, but the data sizes tested were large (11 kB - 5000 kB) in comparison to the 4 bytes typical for sensor readings. There is an exponential correlation between data size and transmission times with data size having negligible effect for smaller data sizes and a much greater effect at large data sizes. There is also a direct correlation between distance and transmission times for large data sizes with negligible effect of distance for smaller data. This paper specifically assesses the effect of distance and data size when sending small data packets consistent with the transmission of sensor data in real time control applications.

Other conclusions from [6] are that concrete walls and metal barriers reduce the effective range of bluetooth to 3 metres, and that transmitting large data, and direct sunlight reduces the effective range to 4-7 metres. Interestingly, wi-fi had no effect on transmission times for data sizes less than 100 kB. Delay variation was measured in [7] and found to be greater than 16% which is less than the industry recommended maximum of 20%. No difference was found in transmitting between mobile phones, pcs or computers.

Some testing [8] was carried out into streaming of MIDI music files via bluetooth with message lengths of between 1 and 6 bytes, with particular emphasis on comparing delays when master and slave device transmit. The result is that the master transmits with mean delay of 30 mS and standard deviation of 10 mS compared to the slave transmitting with mean delay of 20 mS and standard deviation of 20 mS. The discrepancy is occurring because of the different permissions of the master and slave devices in when they can transmit.

II. Android Software Stack

The Android architecture is a software stack comprising applications, a Linux operating system, a runtime environment and various services and libraries. Each layer in the stack and each component in each layer are tightly knitted together to provide a very effective application development and execution platform, as depicted in Fig. 1.

At the bottom of the stack is the Linux 2.6 kernel. Its role is to abstract the hardware into low level software that the higher layers can interact with. It achieves this through hardware device drivers and low level power, process and memory management. Linux is an open source operating system that has been around for decades with widespread application in servers, embedded systems and robotics due to its reliability, efficiency and modularisation of hardware drivers that can be loaded and unloaded while the system is operating.

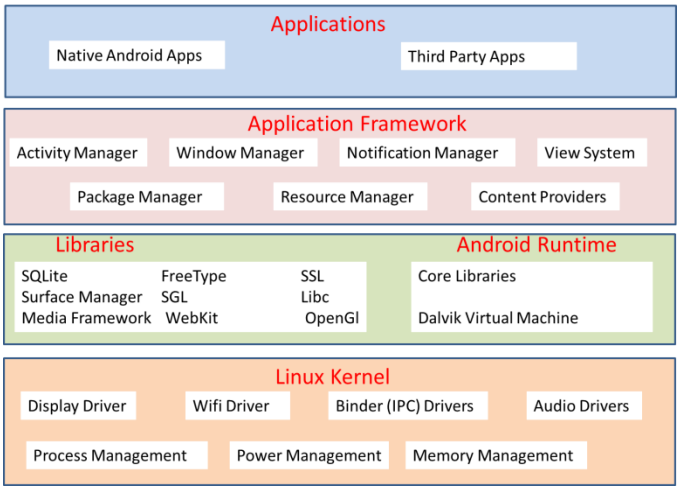


Fig. 1. Android operating system architecture

Each application running on Android is executed on an instance of the Dalvik Virtual Machine. Each application is effectively isolated from every other application, from the operating system and from the hardware device drivers. So each application is developed to run on the Dalvik VM rather than a particular hardware platform. The operation of Dalvik VM is similar to Java VM but more efficient in terms of memory usage and processing power requirements and thus more suitable for smart phones.

The application framework provides already developed support tools for the application while it is running and the basic resources required for an application to run, enabling the developer to program at a higher level. Low level tasks are automated such as the construction, management and end-of-

life clean-up of an activity, the package file structure of an application, and access to common resources. It contains the graphical views that the user would use for the GUI and content providers to share data between processes.

III. BLUETOOTH PROTOCOL

The industrial, scientific and medical (ISM) bands are ranges of radio frequencies reserved internationally for devices that generate electromagnetic emissions that have the potential to cause interference with telecommunication equipment. Devices that can generate electromagnetic emissions, such as microwave ovens, RF heaters and medical diathermy machines, are required to limit their power emissions in these frequency bands. Telecommunication equipment sensitive to electromagnetic interference should avoid these frequencies. However ISM bands have become popular for short-range radio frequency communications like Bluetooth and Wi-fi LAN networks where the potential for interference is limited by their short broadcasting ranges.

The bluetooth channel is a pseudo-random frequency hopping pattern of 79 channels, each with a bandwidth of 1 MHz, within the 2.4 GHz. ISM band, operating between 2.402 – 2.78GHz. The hopping pattern is determined by an algorithm using the address and clock of the master device, to which all devices in the piconet are connected and synchronized. A packet of data will be transmitted on a channel and then each device will switch to the next channel in the frequency hopping pattern before another packet is sent.

Bluetooth incorporates some features to make it more resilient to interference and data loss. Adaptive Frequency Hopping Spectrum is employed to dynamically alter transmission frequencies to avoid frequencies where there is interference. Operating within the ISM band, interference can be expected from other bluetooth devices, IEEE 802.11 WLAN and microwave ovens. The frequency hopping pattern determined by the master device's address and clock can be changed dynamically to avoid frequencies where poor performance due to interference has been detected.

Communication over the channel is serial in nature but parallel communication is achieved by creating time slots to share transmission time, called time-division-duplex (TDD). Each time slot is 625μs in length giving a nominal hopping rate of 1600 hops/sec. Master and slave devices take turns to transmit; the master device transmits in even-numbered time slots and the slave device transmits in odd-numbered time slots. The hop frequency remains constant for the duration of the transmission. When the transmission has completed the channel changes frequency to the next hop frequency in the pattern and the other device transmits.

Large data files will be broken down into packets small enough to be transmitted in one time slot. Each packet consists of a header and payload. The header contains information for channel maintenance and error detection codes and the payload contains user data being transmitted. However packet construction is dynamic where size and composition can be adapted to the conditions. Table II gives a breakdown of the various data packets that can be used. DM1 refers to a small packet designed to be transmitted in one time slot with error detection (FEC) overhead in the header. DH5 refers to a large packet designed to be transmitted in 5 consecutive time slots with no FEC overhead in the header.

TABLE II. PACKET TYPES

Type	Header (Bytes)	Payload (Bytes)	FEC	CRC
DM1	1	0-17	2/3	YES
DH1	1	0-27	NO	YES
DM3	2	0-121	2/3	YES
DH3	2	0-183	NO	YES
DM5	2	0-224	2/3	YES
DH5	2	0-339	NO	YES
AUX1	1	0-29	2/3	NO

Bluetooth employs 3 error detection techniques. FEC1/3 (forward error correction) simply repeats each bit in the header of the packet 3 times. Errors in the header can be easily detected if the bits are not in triplicate and corrected by majority vote. FEC2/3 is a shortened type of Hamming code implemented by appending 5 parity bits to the end of each 10 bit word, making it a 15 bit word. It can correct all single errors and detect all double errors. CRC code (cyclic redundancy check) is used on the data payload in the packet to check it's integrity by referencing the remainder of a polynomial division calculation on the bits in the payload. ARQ (automatic retransmission request) ensures packets will be re-transmitted until an acknowledgement is received from the intended recipient device of a successful, error-free transmission. Error checking overhead can add to transmission delays therefore there is a trade-off between the dual objectives of transmission speed and transmission reliability when using bluetooth.

IV. APPLICATION DESIGN

The application assesses the Android accelerometers and bluetooth module independent of each other. By assessing them independently, their individual contribution to the collective performance when transmitting real-time sensor data wirelessly could be quantified.

Although Android provides the BluetoothAdapter class as an abstraction of the bluetooth hardware, the process of working with bluetooth programmatically is still quite complicated. Because this application was built to assess bluetooth for real world applications, it was decided to build the bluetooth component as a reusable library that could be imported into any future project requiring bluetooth connectivity. Bluetooth operations are effectively simplified to creating an object of BluetoothLibrary and making the correct method calls and interface implementations.

Taking these points into consideration, the application has 3 components;

- An activity to measure the performance of the accelerometers
- An importable BluetoothLibrary to manage the bluetooth connection
- A set of activities to conduct the bluetooth performance assessment

A. *Sensor sampling period testing*

Within the main activity the user can select the sensor testing activity. There are 4 programmable sampling periods for the linear accelerometers. These correspond to the delay in Android sending the sensor data to the application. 'Fastest' corresponds to no imposed delay in sending the reading to the application, 'Game' corresponds to an imposed delay of 20 mS, 'UI' corresponds to an imposed delay of 70 mS and 'Normal' corresponds to an imposed delay of 200 mS. The user can select the sampling period via the radio buttons, Fig.2. By pressing on the 'Begin Test' button the application begins polling the accelerometers. When the test is complete the mean sampling period and standard deviation of the sampling period are posted to the screen. The user can choose to save the results to a csv file in the smartphone's primary memory location.

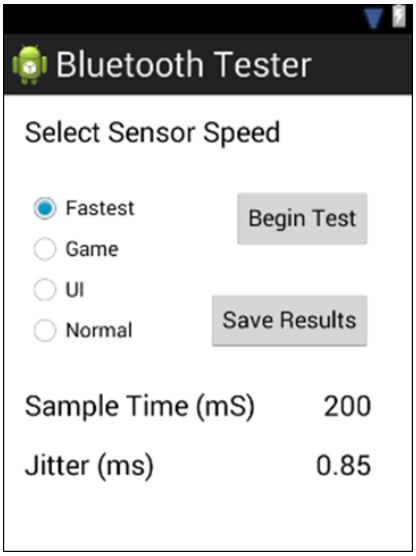


Fig. 2. Sensor sampling period testing activity

Fig.3 is a graphical representation of the operation of the test. Sensor data is received from the onSensorChanged call-back method on the main UI thread and the current time of that event is recorded. The sensor value itself is unimportant for testing, just its timestamp. The timestamp is sent to a parallel thread where all calculations and screen updates are processed in parallel to avoid blocking the callback method in the main thread. This is particularly important when the sampling period is set to 'Fastest' or 'Game' where the GUI can hang due to blocking of the sensor callback. The timestamps from the sensor readings are buffered to avoid overwhelming the run method of the thread. Within the thread calculations are performed to determine the sampling period and a running average of the sampling period is displayed on the screen. Also within the thread the sensor timestamp and period are inserted into a SQLite database for temporary storage and can be saved to the phones memory card for further analysis if the user so wishes. Upon completion of the test the standard deviation of the sampling period (jitter) is calculated by iterating through the SQLite database and displayed on the screen.

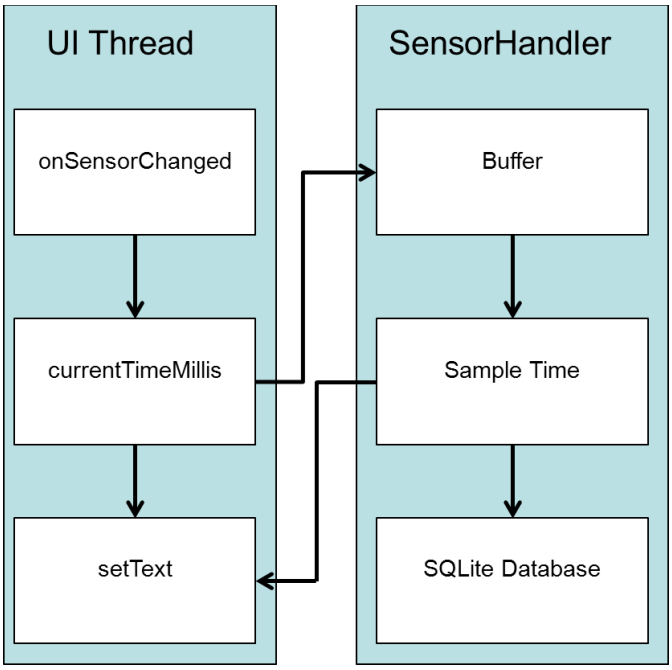


Fig. 3. Operation of the sensor testing activity

B. *Bluetooth Library*

All of the bluetooth related operations that are valid for Android API 8 and that were required for this project are contained within the bluetooth library. This library enables turning bluetooth on and off, making the device discoverable, enabling discovery of other devices, connecting with up to 7 other devices and establishing the input and output streams of a bluetooth connection. Within the bluetooth library is a class, BluetoothLibrary, which contains all of the public methods required for the bluetooth operations. The bluetooth library can be imported into any Android application requiring bluetooth functionality. The structure of the library is shown in Fig. 4.

The connectivity part of the library is contained within its own service. A service in Android is independent of the lifecycle of any activity of the application or the application itself with the advantage that it will allow the established connections to be maintained between activities. Otherwise if the user switches between activities, the activity that established the connection would be paused or destroyed and the connection would be lost.

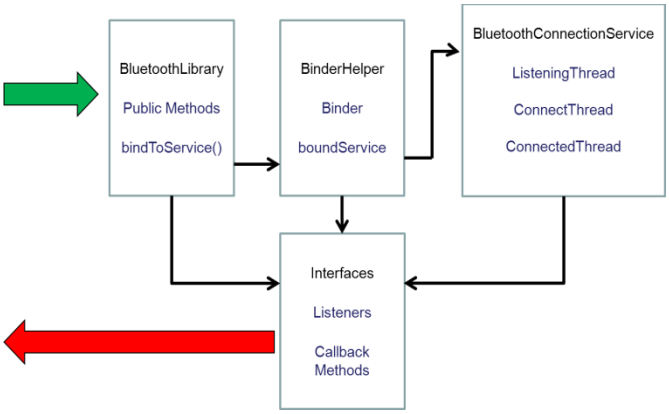


Fig. 4. Structure of the portable library – 'BluetoothLibrary'

The service contains 3 threads, a listening thread which listens for a connection attempt and blocks on the accept method, a connect thread which tries to connect and blocks on the connect method and the connected thread which sets up the input and output streams and blocks on the read method. A typical server operation will listen for a device on the listening thread before switching to the connected thread when it accepts a connection whereas the typical client operation will try to connect on the connect thread before switching to the connected thread to manage the connection.

If a programmer using the library wishes to do anything bluetooth related, they should create an object of BluetoothLibrary within the activity and then call its public methods. If they wish to perform a connection related operation such as checking if a thread is running they will need to bind to the service by calling the bindToService method in the onResume method and unBindFromService method in the onPause method. The call to bindToService is asynchronous which means that the next line of code will be executed before the activity is bound to the service, potentially crashing the application. The programmer can avoid this problem by implementing the onBindListener interface that provides a callback when the activity is bound to the service. Other interfaces are available for turning bluetooth on/off, discovering new devices and receiving data on the input stream.

C. Bluetooth performance testing

The main activity of the application allows all of the normal bluetooth operations to be performed – turning the bluetooth radio on/off, making the device discoverable by other devices, scanning for other devices and initiating a connection. These can be achieved by using the imported bluetooth library. Once two devices are connected, the bluetooth testing activity can begin.

Fig.5 shows the operation of the bluetooth testing activity. One smartphone takes on the role of client and the other smartphone takes on the role of server. The client device sends data to the server device who then returns the data to the client. The client device then calculates the round-trip-time (rtt) for the transmission. This procedure is repeated for 30 seconds until the testing automatically stops.

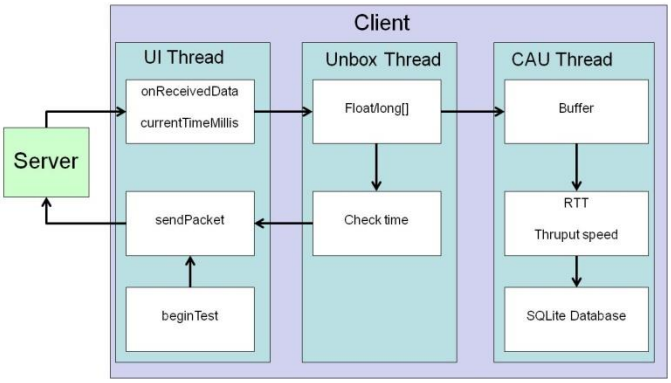


Fig. 5. Operation of the bluetooth performance test

When the transmission of data begins the current timestamp is retrieved from the client system clock. A timestamp is of type long (8 bytes) and it is the timestamp that will be transmitted in the experiment. Depending on the size of data under test, a long array is constructed consisting of the timestamp and filler material, with the exception of the 4 byte data payload size which is tested differently. The data payload is constructed at the beginning of the experiment and sent from client to server and back again in a 30 second loop. Each time the payload is transmitted by the client the current timestamp is retrieved from the client's system clock and inserted into the start of the long array.

Upon receipt of the payload the client retrieves the timestamp and sends it to a thread to perform some calculations, screen updates and data storage while it resends the packet with the new current timestamp. Within parallel threads the round-trip-time is calculated, a running average of the round-trip-time is updated on the screen and the new data is inserted into the SQLite database for storage. The data rate is calculated from the round-trip-time and packet size. Just like in the sensor sampling period test, the network jitter statistic is determined by iterating through the database and calculating the standard deviation of the round-trip-time. The results screen from this part of the application is illustrated in Fig. 6. Multi-threading is used to avoid blocking the main UI thread and slowing the performance of the transmission.

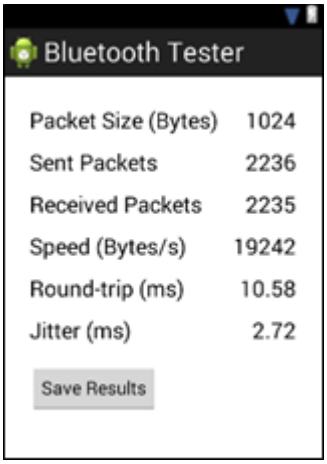


Fig. 6. Client results screen

In the case of the 4 byte test, the long array payload containing the timestamp when the payload was sent is not an option. If the user selects the 4 byte test the payload sent is an auto-incrementing integer which identifies the payload. The timestamp for when the 4 byte payload was sent is recorded in a long variable elsewhere. Apart from that difference, the remainder of the 4 byte test is the same as for the rest of the payload sizes.

V. Testing

The smartphones used in testing were a Samsung S3 running JellyBean and a TCL V860 running GingerBread.

A. Sensor Sampling Period Testing

The procedure for testing the sensor sampling period is straightforward. The user can select one of 4 options for the Android based sampling period. When the test is started a running average of the sampling period is displayed on the screen and upon completion of the test, the standard deviation (jitter) of the sampling period is determined. During tests the running average of the sampling period converges on a value and further testing is not required as the running average will not change significantly from this value. The data stored in the SQLite database is the timestamp of the sensor reading, the period since the previous reading for each sample, a mean sample period and the mean jitter for the experiment as a whole.

B. Bluetooth Testing

The testing of the bluetooth medium was carried out indoors where it is envisaged bluetooth will be used most of the time. When testing bluetooth's performance the factors examined are distance, data payload size and sources of interference. The maximum distance that class 2 bluetooth devices are operable at is 10 meters. Distance between the 2 devices is varied at intervals of 1, 3, 5, 7, and 9 metres. Payload size is varied at intervals of 4, 8, 64, 256, 1024, and 2048 bytes. 4 bytes is the typical size of a sensor reading float value or an integer value and 8 bytes is the size of a long value such as the timestamp.

Testing is carried out in the presence of no interference, an 802.11 wi-fi wireless router and a microwave oven. The testing is carried out for each payload size, at each distance and each source of interference.

VI. Results and Analysis

A. Sensor sampling period testing

The sensor sampling period was tested as outlined in the previous section. The sampling period for all sensor events was stored in the database and graphed for the 4 programmable sampling periods in Android. The mean sampling period was calculated in real time and the jitter was calculated from the sampling periods in the database. The results of testing the 'Normal' (200 mS, 5 Hz) sampling rate are presented in Fig.7.

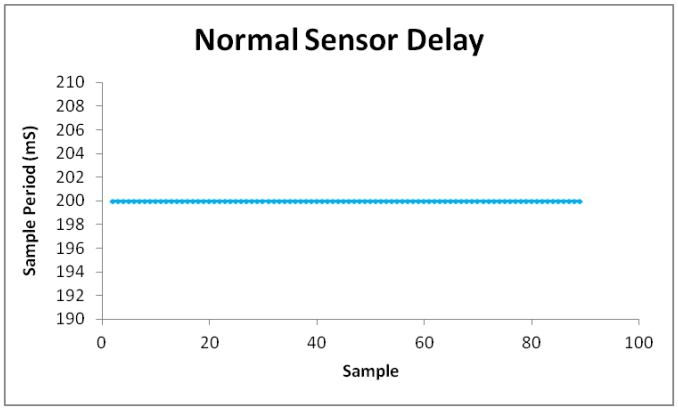


Fig. 7. Performance of 'Normal' sensor sampling rate

'Normal' represents the lowest sampling frequency and longest sampling period which is programmable in Android, 5 Hz and 200 mS respectively. It therefore puts the lowest strain on both the hardware and software. There were approximately 90 readings taken and the performance was consistent for all samples. Jitter was measured at 0 mS in this test. Although the jitter performance of the accelerometer was excellent in this test, there is limited use for such low frequency sampling. Possible uses are measuring the movement of large structures, recording seismic activity and tall building reactions to seismic activity and wind conditions.

Fig.8 presents the results of the 'UI' sensor sampling rate. UI aims to sample the accelerometers every 70 mS (14.3 Hz) which is almost 3 times faster than the 'Normal' sampling rate. From the graph the performance of Android at 'UI' is very good and consistent for the most part. There were 176 sensor samples taken and there were 2 inconsistencies at around sample no. 65 and no. 175.

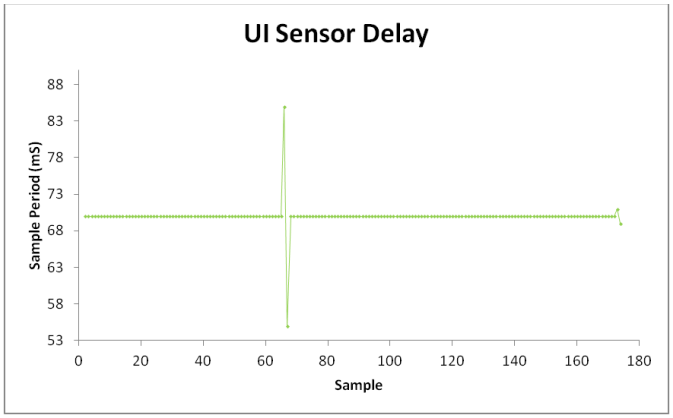


Fig. 8. Performance of 'UI' sensor sampling rate

Sample no. 65 was polled 85 mS after sample no. 64 which is 15 mS late. Also sample no.66 was polled 55 mS after sample no. 65 and 140 mS after sample no.64. This result demonstrates that Android schedules to sample the sensors at

regular time intervals from an initial setpoint, likely to be when the sensorManager is initialized, rather than the previous sensor sample. A similar result is observed at sample no. 175. Android cannot maintain regular sampling of the accelerometer at 14.3 Hz possibly caused by software running in the background using the hardware and operating system resources. It can be deduced that the accelerometers are not given priority by Android when under load. The mean sampling period was 70 mS with a jitter statistic was 0.85 mS in this test.

Understanding this result will aid in understanding the results from testing the 'Game' and 'Fastest' sampling rates in Fig.8. Android claims that 'Game' samples at 20 mS or 50 Hz and that 'Fastest' is limited only by the operating system with no imposed delay. The results in Fig.9 demonstrate the measured performance at these sampling rates.

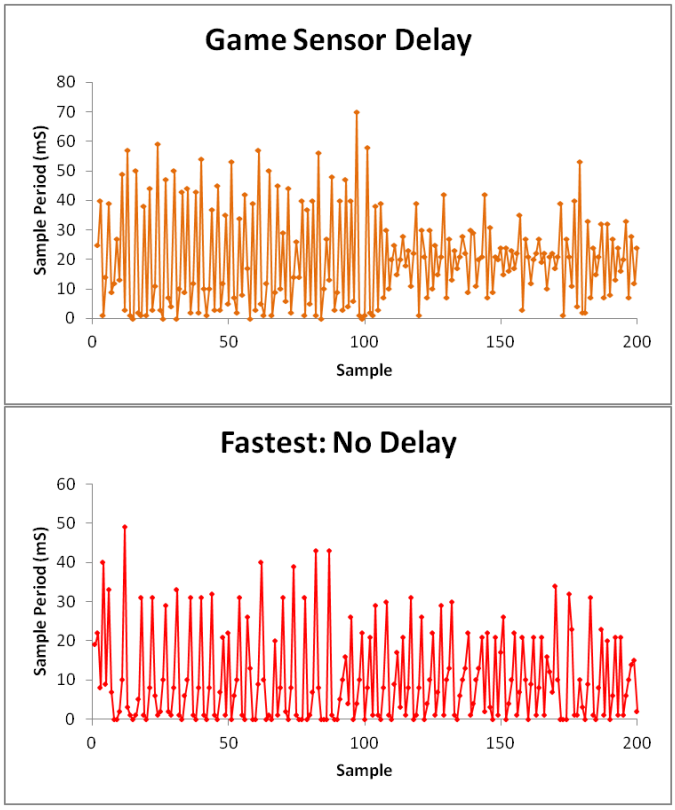


Fig. 9. Performance of 'Game' and 'Fastest' sensor sampling rates

In the case of 'Game', the mean sampling period is calculated to be 20.6 mS, very similar to the nominal value, but with a jitter of 7.8 mS. Sampling periods of 50 mS are not uncommon. Similarly, in the case of 'Fastest', the mean sampling period is calculated to be 5.06 mS with a jitter of 4 mS. Sampling periods of 20 mS are not uncommon followed by 2 or 3 samples taken within a couple of mS of each other. This sampling pattern is repeated throughout this particular test. Sampling period consistency is very poor at the higher sampling rates. The Android operating system and other applications have priority over system resources and interfere with sensor sampling.

B. *Bluetooth Performance*

The performance of bluetooth was tested as outlined in section V. The round-trip-times (rtt) from each experiment were saved in a csv file for analysis and graphing along with a calculation of the mean round-trip-time and standard deviation (jitter).

The effect of distance has on rtt is shown in Fig. 10. It can be seen that, for small data payload sizes, distance has no discernible effect on rtt when tested in an environment with no interference. However in the case of the two larger payload sizes, there is a step change of 10-14 mS in rtt performance improvement between 3 and 5 metres. At the application level it is not immediately obvious what low level bluetooth changes occurred to cause this step change. However bluetooth is a dynamic wireless transmission protocol, continuously changing frequencies, packet size and error correction overhead to improve performance. It is possible that in the case of data greater than 1 kb transmitted over distances greater than 3 metres that larger data packets were used, perhaps a 5 slot packet rather than a 3 slot packet used for smaller data sizes.

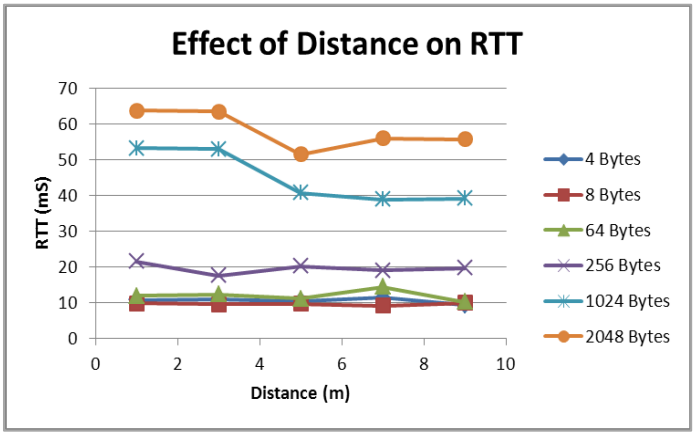


Fig. 10. Effect of distance on round-trip-time

Fig.11 shows the effect of payload size on rtt where it is apparent there is a strong correlation between payload size and rtt. Note the 7 metre curve is partially obscured by the 9 metre curve. Unexpectedly, the rtt performance of bluetooth is poorer at 1 and 3 metres for data payload sizes greater than 1 kb. The relationship appears to be non-linear but there is not a sufficient range of payload sizes to fully model the trend. The overhead induced by increasing payload appears more profound over the shorter distances. Larger data payloads require a greater number of packet transmissions. This effect is non-linear due to the dynamic nature of bluetooth packet assignment.

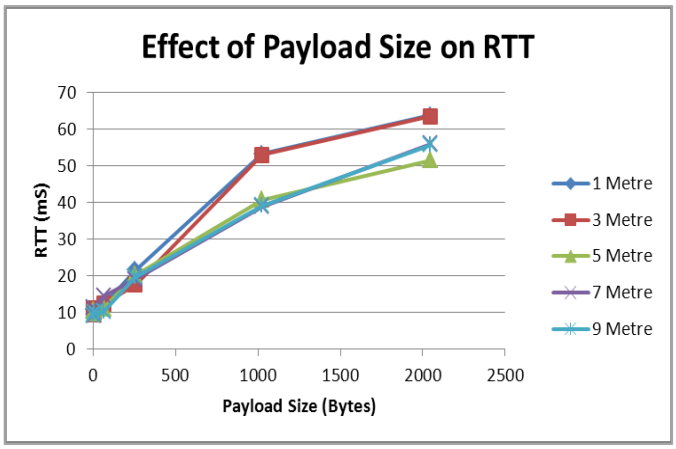


Fig. 11. Effect of data payload size on round-trip-time

One important result from Fig.11 is the offset on the rtt axis. Given that payloads of 4 and 8 bytes were tested, which are almost the smallest payload possible (only char is smaller at 2 bytes in size), there appears to be a minimum payload rtt overhead of approximately 10 mS per payload at the Android application level regardless of payload size. At the Android application level it will take the operating system a minimum of 10 mS to process the outgoing data and the incoming data for each payload in a round-trip scenario. The consequence of this result is that, in the case of round trips, the maximum payload frequency is limited to 100 Hz and an estimated 200 Hz in the case of a one-way transmission. This result suggests that Android is more suitable for single large data file transmission rather than multiple small packets.

From calculations of data rate, with payload size of 2048 bytes the data rate is in the region of 30-35 kB/s whereas for the 4 byte payload the data rate is approximately 800 bytes/s. Bluetooth is rated at 2.1 Mb/s and that may be possible when transmitting a single very large data payload. From Android's perspective most bluetooth users would be using bluetooth for transmitting large data files rather than bursty data of small size. This effect can also be seen in using bluetooth to stream audio where a noticeable latency can be detected.

The effects of wi-fi and microwave interference on rtt and jitter are shown in Fig.12. IEEE 802.11 Wi-Fi and microwave ovens operate within the same ISM band as bluetooth and have been identified as potential sources of interference. Microwave ovens operate at a fixed frequency of 2.45 GHz. IEEE 802.11 operates between 2.4 and 2.5 GHz and uses Direct Sequence Spread Spectrum (DSSS) to avoid interference.

From the graphs, only microwave interference appears to have an effect on rtt and an increasing effect for larger data payload sizes. There is a consistent jitter of 3-4 mS regardless of interference source or no interference which is to be expected from unprotected wireless transmission through the air.

However inconsistencies appear in the graph with wi-fi performing better than the interference free case in the rtt test which is unexpected. Further inconsistencies are apparent in the jitter graph where surprisingly wi-fi and microwave have more consistent round-trip-times than the interference free test.Interference by its nature is non-homogenous and inconsistent. Furthermore the interference free test is free of obvious sources of electromagnetic interference but it is not free of background radiation in the air. Therefore bluetooth's performance will be inconsistent and unpredictable dependent on the conditions of the environment in which it is operating.

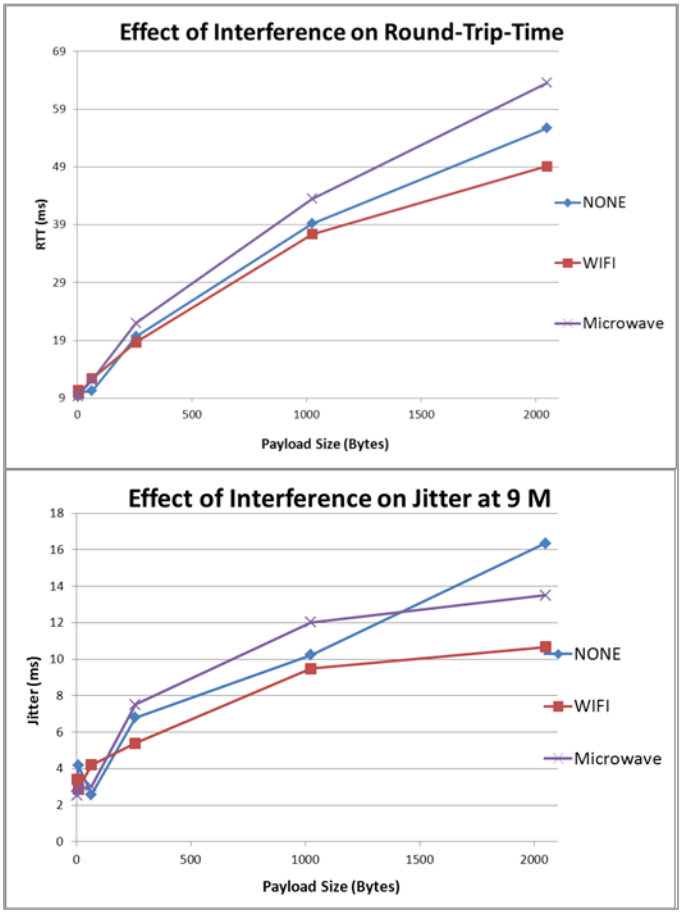


Fig. 12. Effect of interference on jitter at a range of 9 metres

The previous results in the interference tests were inconclusive due to the inconsistent nature of interference and bluetooth's adaptation to the transmitting environment and so further analysis of the data from the microwave oven test was carried out. Fig.13 shows the effects of microwave interference on rtt and jitter. Rtt is largely unaffected by distance from a microwave source. Previous results in Fig.10 had shown that distance alone had no effect on rtt. The jitter graph shows that microwave increases jitter when both smartphones are within 1-3 metres of the source.

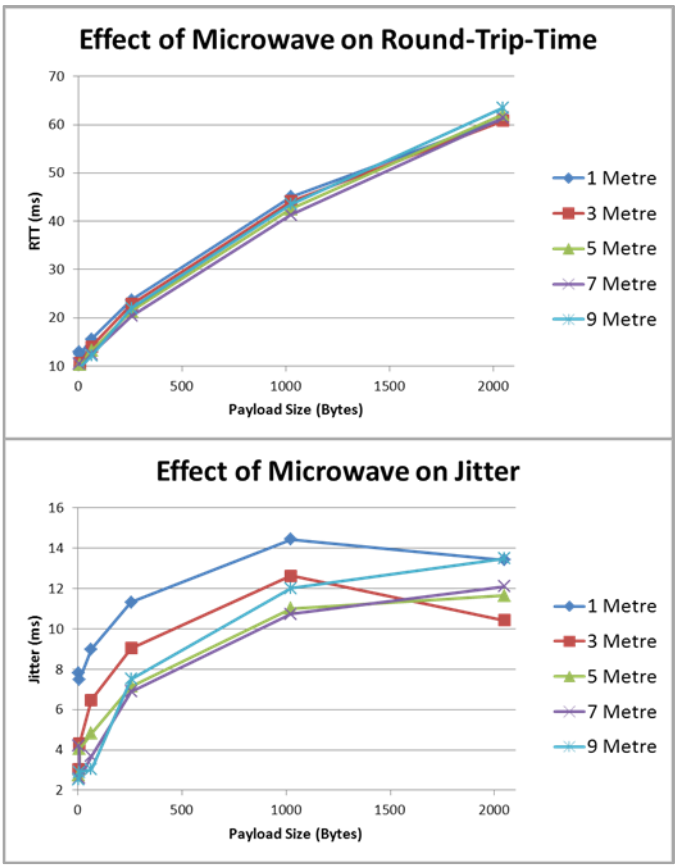


Fig. 13. Effect of microwave interference on round-trip-time and jitter at various distances

VII. Conclusion

This paper has determined that Android bluetooth is geared towards sending single large data files such as music or video or document sharing rather than high frequency small discrete pieces of data such as sensor readings. Bluetooth can transmit sensor readings of up to 64 bytes at 100 Hz round-trip, or 200 Hz one way. Distance has no effect on transmission time for a class 2 device within the 10 metre range. Transmission time increases non-linearly with increasing data size. Bluetooth is resilient to microwave and wi-fi interference.

The on-board accelerometers can only be consistently sampled at 14.3 Hz. The maximum mean sampling frequency is 200 Hz but with a standard deviation of 80%. The Android accelerometers are geared towards screen orientation and game play with low system priority given to the on-board sensors.

In terms of utilising Android smartphones' onboard sensor and bluetooth technology for real-time control applications, from the results of testing in this paper, it is estimated that the sensors reliable sampling limit is 14 Hz and that the sensors' output can be transmitted via bluetooth to another device within 5 mS over a range of 10 metres. The effects of distance and interference can be neglected.

The Android system sets up its bluetooth radio and accelerometers for the functionality it has deemed most useful for its users, file sharing and game play. Although Android's sensors and bluetooth radio are not suitable for most real-time control applications, quantification of the performance of Android in this paper may prove useful to readers in their own projects.

References

[1] [Online], "Developer Economics Q3 2013 analyst report", http://www.visionmobile.com/DevEcon3Q13, accessed December 2014

[2] [Online], "Android API", http://developer.android.com/guide/topics/sensors/sensors_overview.html, accessed December 2

[3] Andersson M., Bluetooth For Industry, The Industrial Ethernet Book, 11 (September 2002), pp. 5-11.

[4] [Online], "The Bluetooth Special Interest Group," http://www.bluetooth.com, accessed November 3013

[5] [Online], Garcia Pique, J., Lozano Almazan, I., Sanchez Garcia, D., web.udl.es/usuaris/carlesm/docencia/xc1/Treballs/Bluetooth.Treball.pdf, accessed March 2014

[6] Pudaruth, S., Ramdolin, H.K., Bissoonee, A., "An assessment of the performance of bluetooth as a broadcasting channel", Proceedings of the World Congress on Engineering 2010 Vol IWCE 2010, June 30 - July 2, 2010, London, U.K.

[7] Rashid, R.A., Yusoff, R., "Bluetooth performance analysis in personal area network", Proceedings of the 2006 International RF and Microwave Conference, September 12 - 14, 2006, Putrajaya, Malaysia

[8] Bartolomeu, P., Fonseca, J.A., Duarte, P., Rodrigues, P.M., Girao, L.M., "MIDI over Bluetooth", Proceedings of the Conference on Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE, Volume: 1

Design of a Cloud Learning System Based on Multi-Agents Approach

Mohammed BOUSMAH, Ouidad LABOUIDYA, Najib EL KAMOUN
STIC Laboratory, Faculty of Science
Chouaib Doukkali University
El Jadida, MOROCCO

***Abstract*—Cloud Computing can provide many benefits for university. It is a new paradigm of IT, which provides all resources such as software (SaaS), platform (PaaS) and infrastructure (IaaS) as a service over the Internet. In cloud computing, user can access the services anywhere, at any time and using any devices (Smart phones, tablet computers, laptops, desktops…). Multi-Agents System approach provides ideal solution for open and scalable systems whose structure can be changed dynamically. Educational institutions all over the world have already adapted the cloud to their own settings and made use of its great potential for innovation. Based on the analysis of the advantages of cloud computing and multi-agents system approach to support e-learning session, the paper presents a complete design and experimentation of a new layer in cloud computing called Smart Cloud Learning System.**

Keywords—Cloud computing; Multi-Agents System; Project Based Learning

I. INTRODUCTION

Smart Cloud Learning System is an hybrid approach that combine the Cloud Computing, the Multi-Agents Technology and the Learning Management System. In fact, Cloud Computing has become a popular topic in the research community because of its ability to transform computer software, platforms, and infrastructure as a service. Multi-Agents System approach provides ideal solution for open and scalable systems whose structure can be changed dynamically and asked cooperation, interaction and negotiation. Online learning is now a reality thanks to the development of Internet and to the virtual environments commonly called LMS (Learning Management System). But, to support the interactions of the various actors intervening in the formation (learner, tutor, teaching designer, coordinator...) and to propose the data processing tools and artefacts for their giving a support and assistance, constitutes today a serious problems, renewed recently by the explosion of the research on the e-learning.

In this paper, we present a complete design and experimentation of a new layer in cloud computing called Smart Cloud Learning System.

The first part of this paper introduces the design of a cloud learning system based on multi-agents approach called Smart Cloud Learning System (Smart-CLS). The second part presents the experimental results and discussion. We finish by the conclusion and the future work.

II. FROM CLOUD COMPUTING TO SMART CLOUD LEARNING SYSTEM (SMART-CLS)

Currently in literature, we can find several definitions for the cloud computing. According to the National Institute of Standards and Technology [1], "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics (On-demand self-service, Broad network access, Resource pooling, Rapid elasticity, Measured Service); three service models (Cloud Software as a Service (SaaS), Cloud Platform as a Service (PaaS), Cloud Infrastructure as a Service (IaaS)); and, four deployment models (Private cloud, Community cloud, Public cloud, Hybrid cloud). Key enabling technologies include: (1) fast wide-area networks, (2) powerful, inexpensive server computers, and (3) high-performance virtualization for commodity hardware".

Cloud computing provides a scalable online environment that makes it possible to handle an increased volume of work without impacting system performance. In our sense, cloud computing can provide many benefits for university:

- Lower capital costs: University can provide unique services using large-scale computing resources from cloud service providers, and then nimbly add or remove IT capacity to meet peak and fluctuating service demands while only paying for actual capacity used.
- Lower IT operating costs: University can rent added server space for a few hours at a time rather than maintain proprietary servers without worrying about upgrading their resources whenever a new application version is available. They also have the flexibility to host their virtual IT infrastructure in locations offering the lowest cost.
- No hardware or software installation or maintenance
- Optimized IT infrastructure provides quick access to needed computing services
- User can access the services anywhere, at any time and using any devices.

In these advantages, many researchers of e-learning area [2] [3] [4] [5] [6] attempt to apply their process to cloud computing. It is one of the new technology trends likely to have a significant impact on the learning environment in recent years. However, the data privacy and security are the main risk[1] [7].

Our proposal revolves around three elements:

- Cloud computing must support e-learning and m-learning such as LaaS (Learning as a Service);
- Multi-Agents technology must be integrated in Cloud computing as a service;
- A new layer called Smart Cloud Learning System (Smart-CLS) must be set, in order to provide services anywhere at any time and using any devices, for all actors in learning session.

Founded on these ideas, we propose the architecture shown in "Fig. 1". Smart-CLS is a layer of software that creates a common platform for all communications human-to-system, Web-based, and mobile-device-based interactions. Smart-CLS has two major benefits:

- Provide a Graphical User Interfaces agent (GUIs agent)
- Facilitate the deployment of multi-agent system in the cloud.

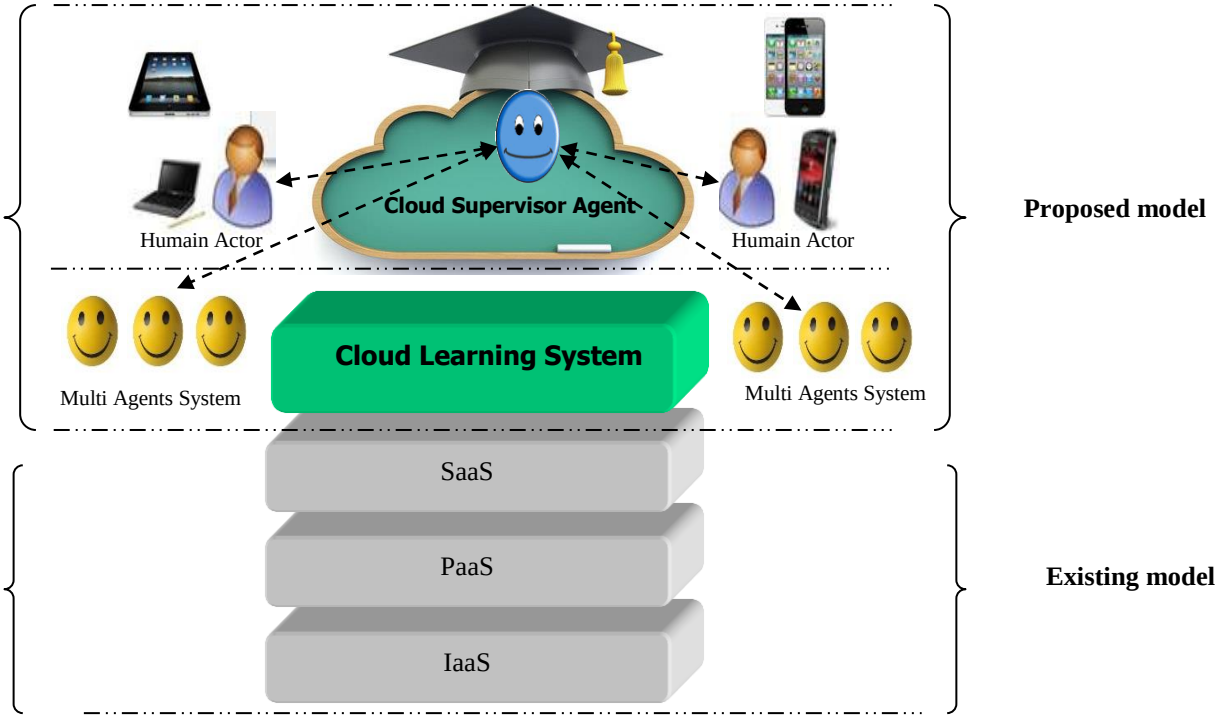


Fig. 1. Smart Cloud Learning System Architecture

[1] http://www.claconnect.com/Risk-Management/The-Benefits-and-Risks-of-Cloud-Computing.aspx

Cloud infrastructures can offer an ideal platform where run Multi-agents system [8]. In fact, an agent is a software entity capable of acting intelligently on behalf of a user, in order to accomplish a given task. Agents, like humans, co-operate so that a society of agents can combine their efforts to achieve a desired goal [9]. The characteristic properties of the agents are:

- Autonomy
- Proactive intelligence (agents do not simply to act in response to their environment, but are able to take initiative)
- Temporal continuity (they are continuously running processes),
- Mobility,
- Rationality/benevolence (agents don't have conflicting goals)
- Adaptive intelligence (agents have the ability to learn).

Compared to objects, software agents have their own thread of control, localizing not only code and state but their invocations as well. In other words**,** agents themselves define when and how to act.

Agent-oriented methodologies and platforms have become a priority for the development of large scale agent-based systems. Several methodologies have been proposed for the development of multi-agent systems (MAS), they are either an extension of object-oriented methodologies (for example MaSE: Multi-agent System Engineering) [10] or an extension of knowledge-based methodologies (for example: CommonKADS) [11].

We have chosen the MaSE methodology (Multiagent System Engineering) for the development of our software agents. This choice is justified by:

- The simple, modest and pragmatic vision which MaSE gives to the definition of an agent
- The automation process for creating software agents
- The availability of documentation.

The systems based on agents specified starting from this methodology are often difficult to implement directly starting from the standard programming languages like Java or others. Several tools are developed recently for multi-agent systems programming: JADE [12], Zeus [13], MadKit [14], AgentBuilder [15]. For our part, after an evaluation of the most popular platforms of multi-agent systems development, we have chosen JADE (Java Agent Development Framework) which is a middleware that facilitates the development of multi-agent systems. It includes:

- A runtime environment where JADE agents can "live" and that must be active on a given host before one or more agents can be executed on that host.
- A library of classes that programmers have to/can use (directly or by specializing them) to develop their agents.
- A suite of graphical tools that allows administrating and monitoring the activity of running agents.

Basing on our model of a virtual campus [16] and on an approach centred on the roles and competences, we can specify and identify the agents which will build our Smart Cloud Learning System (Smart-CLS). This process is located in an iterative step of design whose results presented here are those after the most recent iteration. We present in "Fig. 2", an observer system of use for a given space in Smart Cloud Learning System.

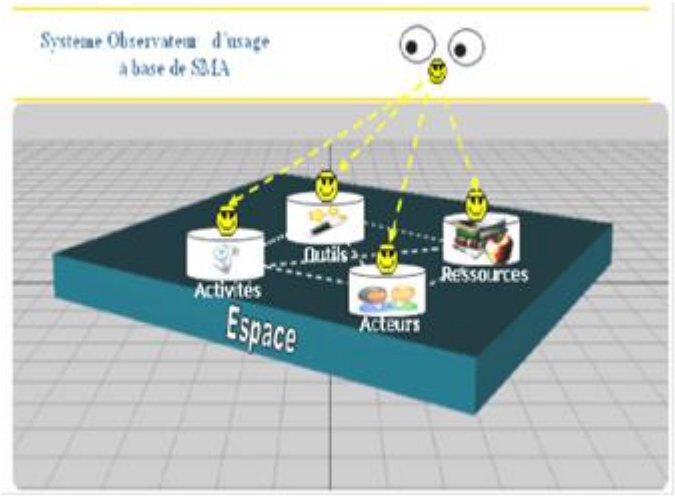


Fig. 1. Smart -CLS observer system of use

The agent hierarchy of our Smart-CLS reporting system is made around several supervisory spaces agents. In occurrence public space, the group space, the team space and individual space. Each supervisory space agent communicates with four agents: the supervisory agent of the actors, the supervisory agent of the activities, the supervisory agent of the resources and finally the supervisory agent of the tools. Each one of these four agents can supervise other agents of lower hierarchy. For example, the supervisory agent of the tools can supervise the mail agent, the forum agent, the discussion agent, the document agent and the diary agent. Finally, a graphic user interface agent must ensure the communication with the user. We summarize here, the specifications of the agents constituting our reporting system in Smart-CLS:

- Graphical User Interface Agent (GUI Agent): its role is to ensure the human/machine communication trough a simple and convivial graphic interface.
- Supervisory Agent of space (public, group, team, individual): This agent is the access point to the space of which it is monitor. It answers the lower hierarchy agents for a reporting of a given period of use.
- Supervisory Agent of the actors: It supervises the whole of the actions carried out by an actor while providing a decision on its behaviour during a training session.
- Supervisory Agent of the activities: It indicates the degree of project respect and the level of activity success.

- Supervisory Agent of the tools: Its role is to provide statistics concerning the use of the tools with a relation of a given space.
- Supervisory Agent of the resources: It gives information on the use of the resources of a given space.

III. Smart Cloud Learning System: Experimental Results

A. Context and Objective

Recent years have seen a growing recognition and general acceptance of the Project Based Learning (PBL) in education, especially in Moroccan higher education. This approach transforms teaching from "teachers telling" to "students doing" [17] [18] [19]. Students are facing several challenges during a project based learning session. The biggest challenge is: What methodology should be followed for successful project? In our laboratory, we propose to use the project management process such us a methodology that should be followed for successful project. In this context, we have conducted two experiments.

The objectives of these experiments are to understand the benefits of Smart-CLS in project management process and the feedback of students about it.

The first experiment has focused on a group of 69 undergraduate students in project based learning session without Smart-CLS during four weeks of May 2012 [20].

The second experiment has focused on a group of 109 undergraduate students in project based learning session with Smart-CLS, via a simple Web browser and an Internet connection, during four weeks of May 2014. Six Graphical User Interfaces agents (GUIs agents) of Smart-CLS have been used as avatar in order to facilitate the use of project management tools “Fig. 3”; namely:

- FA-Agent (Functional Analysis Agent), used to help students in Functional Analysis Phase
- FAST Agent (Functional Analysis System Technique Agent), which has a strict translation of each of the service functions in technical(s) function (s), then the constructively (s) solution (s)
- WBS Agent (Work Breakdown Structure Agent), which allows the cutting of the project task list
- RACI Agent (Realization, Approval, Consulting, Information Agent), which allows the definition of responsibilities.
- PERT Agent (Program Evaluation and Review Technique Agent) used to schedule, organize, and coordinate tasks within a project.
- GANTT Agent used for planning and scheduling projects.

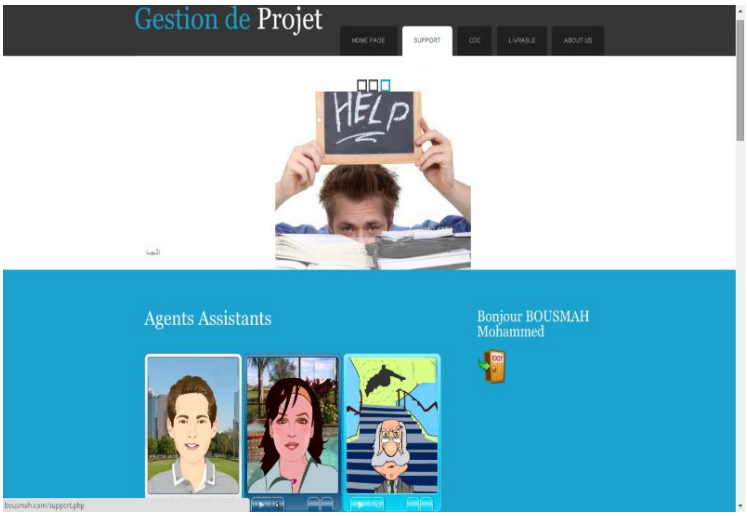


Fig. 2. Smart-CLS with these GUIs agents

B. Result and discussion

At the first experiment, and as shown in figure 4, we have found a low percentage of use of project management tools. In fact, 34% of students have developed the functional analysis chart, only 20% have used the FAST diagram, 47% have broken the project into tasks WBS diagram, 24% have developed the RACI matrix, 19% have used the PERT diagram, 35% have used GANTT diagram for planning and scheduling projects.

In our sense, this result can be explained by the fact that the project management approach itself, consist of complex tasks that pose problems for students. Therefore, help and assistance is necessary. Hence the use of Smart –CLS.

At the second experiment,

- First, as shown in “Fig. 4”, we observe a high percentage of use of project management tools. This result demonstrates the utility of Smart-CLS in this process. It is some of the most important applications of cloud computing in our University, with an aim to provide help and assistance as a modern ways of learning and teaching.
- Second, as shown in “Fig. 5”, we note the type of devices used to access to Smart-CLS, in fact, 73% of students have used a laptop, 8% have used a desktop, only 2% have used a smartphone or tablet and 17% have accessed via an Internet cafe. However, with the increased availability of low-priced tablets and smartphones, it is expected that even more students would get an opportunity to get access to these tools over the next few years.

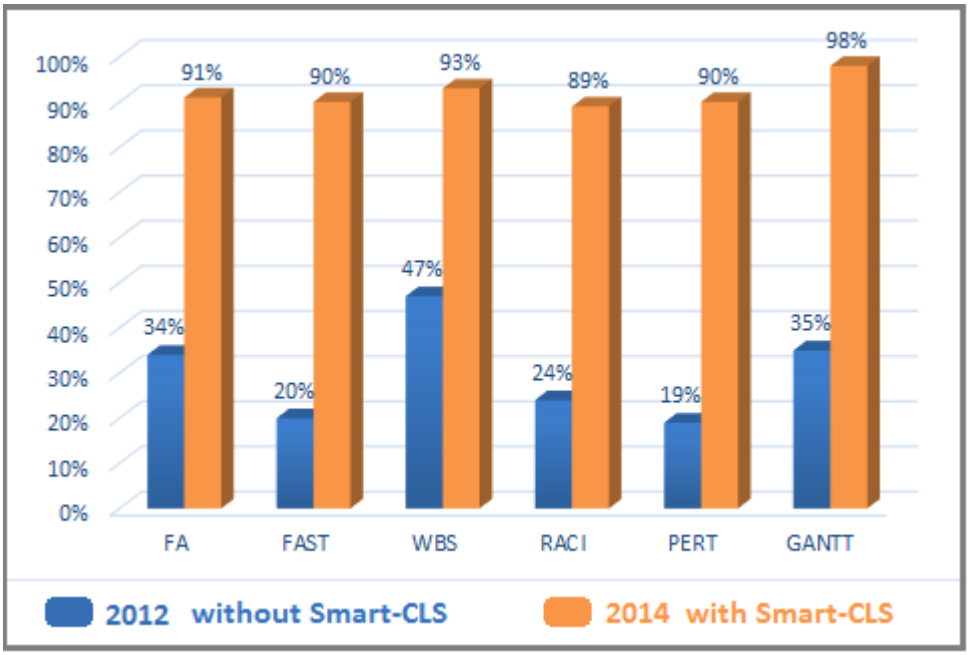


Fig. 3. Statistic of project management tools used by students

- Third and as shown in "Fig. 6", we observe a high percentage of agent consultation rate, we find that the GUIs agents have been very consulted by students, in order to give them help and assistance; this can be explained by the difficulties encountered by students to implement the project management process.
- Finally, "Fig. 7" shows the results of a assessment questionnaire offered to students at the end of the second experiment. The objective of this questionnaire was to evaluate the quality of help and assistance provided by GUIs Agent. Students are most satisfied with these agents. It is expected that even more students would get an opportunity to get access to these tools over the next few years, it is a great potential for innovation.

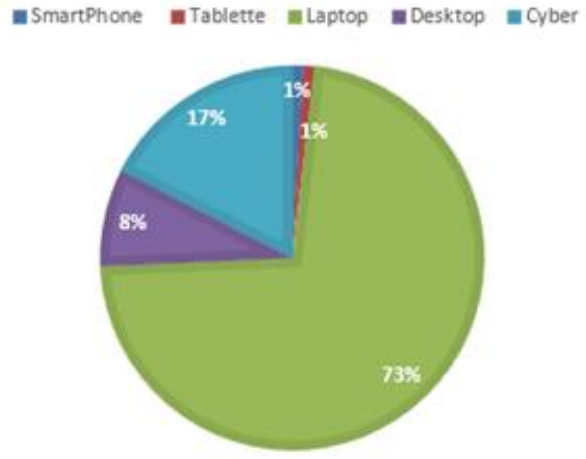


Fig. 4. Devices used for access to Smart-CLS

Fig. 5. Rate of GUIs Agent consultation by students

Fig. 6. Quality of help and assistance provided by GUIs Agent

IV. Conclusion & Future Work

We have demonstrated in this article that cloud computing can provide many benefits for university, and we have proposed an hybrid approach that combine the Cloud Computing and the Multi-Agents Technology to design a new layer called Smart Cloud Learning System.

Experimental results in a project management process context have validated the use of Smart-CLS and have demonstrated that it is some of the most important applications of cloud computing in our University, with an aim to provide help and assistance as a modern ways of learning and teaching. Students have been most satisfied.

The future work consists of experiment and validates Smart-CLS in Massive Open Online Courses (MOOCs). It is expected that even more students would get an opportunity to get access to these tools over the next few years, it is a great potential for innovation.

REFERENCES

[1] NIST, National Institute of Standards and Technology. 2013. " Cloud Computing Standards Roadmap Working Group", U. S. Department of Commerce, Reports on Computer Systems Technology, July 2013

[2] Whitepaper by Crucial Cloud Hosting, Cloud Computing in Education, March, 2014, available from http://www.crucial.com.au/pdf/Cloud_Computing_in_Education.pdf [Accessed 4th February 2015]

[3] Nungki Selviandro, Zainal Arifin Hasibuan. 2013. "Cloud-Based E-Learning: A Proposed Model and Benefits by Using E-Learning Based on Cloud Computing for Educational Institution", Springer Information and Communication Technology, Volume 7804, 2013, pp 192-201

[4] Md. Anwar Hossain Masud, Xiaodi Huang, "An E-learning System Architecture based on Cloud Computing," International Journal of Social and Human Sciences 04/2012; 6.

[5] E. Tuncay, "Effective use of Cloud computing in educational institutions, " Procedia Social Behavioral Sciences, p. 938–942, 2010.

[6] Y. Zhongze, "The basic principles of cloud computing and its impact on education", Satellite TV and Broadband Multimedia, 2010.6, pp.67-70.

[7] Z. Chengyun, "Cloud Security: The security risks of cloud computing, models and strategies", Programmer, May.2010, pp.71-73.

[8] Domenico Talia, "Cloud Computing and Software Agents: Towards Cloud Intelligent Services, " Proceedings of the 12th Workshop on Objects and Agents, Rende (CS), Italy, Jul 4-6, 2011, pp.2-6.

[9] Wooldridge Michael and Nick Jennings, "Intelligent Agents: Theory and Practice", Cambridge University Press , 1995.

[10] Arnon Sturm. 2005. "Multiagent Systems Engineering (MaSE) – An Introduction". 12-Jul-2005, available from http://www.pa.icar.cnr.it/cossentino/al3tf1/docs/mase4agentlink.pdf [Accessed 4th February 2015]

[11] C. e.a. Iglesias, "Analysis and design of multiagent systems using MAS-CommonKADS," in Proc. of AAAI'97, Workshop on Agent Theories, Architectures and languages, Providence, RI, 1997.

[12] Rimassa G., Bellifemine F., Poggi A, "JADE - A FIPA Compliant Agent Framework". PMAA`99, p. 97-108, Londres, Avril 1999.

[13] L. C. Lee, D. T. Ndumu, and H. S. Nwana , " ZEUS: An Advanced Tool-Kit for Engineering Distributed Multi-Agent Systems," in Proceedings of the Practical Application of Intelligent Agents and Multi-Agent Systems, p. 377-392, Londres,1998.

[14] O. Gutknecht, J.Ferber & F. Michel, RR, "MadKit: une plateforme multi-agent générique". Rapport interne, Laboratoire LIRMM, Université Montpellier II, Mai 2000

[15] AgentBuilder U.G.2004. An Integrated Toolkit for Constructing Intelligent Sofware Agents, AgentBuilder, User's Guide, Avril 2004, available from http://www.agentbuilder.com/Documentation/UsersGuide-v1.4.pdf [Accessed 4th February 2015]

[16] N. Elkamoun, M. Bousmah, Abdelhak Aqqal, A. Berraissoul. "Conception et réalisation d'un environnement virtuel d'apprentissage collaboratif, orienté métaphore spatiale, couplé avec un système observateur d'usage, " International electronic journal of the Information Technologies e-TI, vol. 2, no. 2, April 2006. ISSN 1114-8802

[17] M. Bousmah, N. Elkamoun, A. Berraissoul," Online Method and Environment for Elaborate the Project-Based Learning Specifications in Higher Education", Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies, ICALT 2006, 5-7 July 2006, Kerkrade, The Netherlands. IEEE Computer Society 2006 BibTeX.

[18] Najib EL KAMOUN, Mohammed BOUSMAH, Abdelhak AQQAL. 2011. "Virtual Environment Online for the Project-Based Learning Session ", Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Software Engineering (JSSE), January Edition, 2011.

[19] Clifford F. Gray & Erik W. Larson. 2006. " Project Management the Managerial Process". (3rd ed.) McGraw Hill. pg. 6

[20] M. Bousmah, N. Elkamoun. 2014. " Feedback of Project Management Approach in Higher Education Science" 26th Symposium of the European ADMEE 15-16-17 January 2014 in Marrakech, Morocco.

Standard Positioning Performance Evaluation of a Single-Frequency GPS Receiver Implementing Ionospheric and Tropospheric Error Corrections

Alban Rakipi [1], Bexhet Kamo [1]
[1] Faculty of Information Technology
Polytechnic University of Tirana
Tirana, ALBANIA

Shkelzen Cakaj [1,2], Algenti Lala [1]
[2] Faculty of Electrical and Computing Engineering
Prishtina University
Prishtina, KOSOVO

***Abstract*—This paper evaluates the positioning performance of a single-frequency software GPS receiver using Ionospheric and Tropospheric corrections. While a dual-frequency user has the ability to eliminate the ionosphere error by taking a linear combination of observables, a single-frequency user must remove or calibrate this error by other means. To remove the ionosphere error we take advantage of the Klobuchar correction model, while for troposphere error mitigation the Hopfield correction model is used. Real GPS measurements were gathered using a single frequency receiver and post–processed by our proposed adaptive positioning algorithm. The integrated Klobuchar and Hopfield error correction models yeild a considerable reduction of the vertical error. The positioning algorithm automatically combines all available GPS pseudorange measurements when more than four satellites are in use. Experimental results show that improved standard positioning is achieved after error mitigation.**

Keywords—algorithm; Global Positioning System; GDOP; Hopfield model; Klobuchar model; receiver; PVT; Raw measurements

I. INTRODUCTION

Recently, there is an increase interest in positioning techniques based on GNSS (Global Navigation Satellite Systems) such as GPS (Global Positioning System). GPS is a satellite-based navigation radio system which is used to verify the position and time in space and on the Earth [1]. The standard approach for estimating the receiver position and clock offset is first to linearize the pseudorange measurements around a rough guess of the receiver position and clock bias and then to iterate until the difference between the guess and the measurements approaches zero. While this implies that some information is needed about the initial receiver position, it turns out that the solution is not very sensitive to this initial (or rough) guess [2]. The GPS satellites are orbiting the Earth at altitudes of about 20.200 km and it is generally known that the atmospheric effects on the GPS signals are the most dominant spatially correlated biases. The atmosphere causing the delay in GPS signals consists of two main layers: ionosphere and troposphere [3].

The Ionosphere is the band of the atmosphere from around (50 – 1000 km) above the earth's surface and is highly variable in space and time, with certain solar-related ionospheric disturbances [4]. Ionosphere research attracts significant attention from the GPS community because ionosphere range delay on GPS signals is a major error source in GPS positioning and navigation. The ionospheric delay is a function of the total electron content (TEC) along the signal path and the frequency of the propagated signal, mostly affecting the vertical component of user's position. Two main statistical model are available for the correction of ionospheric range error in single frequency applications: the Klobuchar model for GPS [3] or the NeQuick model [2] foreseen for use in European GALILEO system.

The troposphere is the band of the atmosphere from the earth's surface to about 8 km over the poles and 16 km over the equator [5]. The tropospheric propagation delay is directly related to the refractive index (or refractivity). The signal refraction in the troposphere is separated into two components: the dry and the wet component, where the dry or hydrostatic component is mainly a function of atmospheric pressure and gives rise to about 90% of the tropospheric delay. There are different mathematical models that can be used to correct the tropospheric error such as Saastamoinen and Hopfield Model [6].

The paper is organized into seven major sections. The first section goes over background on positioning techniques based on GNSS and atmospheric errors. The second section describes the data collection process and the tools used for measurements. The third section is referred to acquisition and tracking of GPS signals. The fourth section gives a high level description of our approach in implementing the positioning algorithm. Section five is dedicated to Ionospheric and Tropospheric error correction models focusing on Klobuchar and Hopfield models. Section six presents some results obtained analyzing the algorithm performance with and without error corrections. Finally the last section draws the conclusions.

II. DATA COLLECTION PROCESS

In this section is described the GPS data collection process and the implementation of a post-processing adaptive Position Velocity Time (PVT) algorithm, where we included mathematical Ionospheric and Tropospheric correction models aiming to an improved accuracy of user's position estimation. An experiment was conducted using GPS C/A-code pseudorange data collected outside our laboratory in the

Polytechnic University of Tirana Campus. The precise Cartesian coordinates of our stationary receiver were previously determined by a professional receiver for later comparison. Totally 2340 epochs of data were analyzed for the experiment and post-processed in Matlab® environment. In the following subsections we describe the tools used for measurements.

A. Receiver Unit

In this section we give a brief description of the receiver used to collect the data and of the software used to process them. SAT-SURF [7] is a hardware black-box integrating GPS and GSM/GPRS functionalities. It appears as a metallic box with two external antennas, a USB cable and a power supply cable (Fig. 1). The SAT-SURF hardware allows getting out from the GPS receiver several data and also each available raw measurement (depending on the receiver capabilities). Each GPS parameter is logged with a related GPS time stamp, so that each parameter can be aligned to the evolution of all the others. In our measurements we used SAT-SURF with the core components indicated in Table 1.

SAT-SURFER [7] is the software suite running on a standard PC that uses data coming from SAT-SURF. It is a software suite able to talk in real-time with state-of-the-art GPS receiver modules as well as external professional GPS units. SAT-SURFER gets raw data, displays such data on the screen and log them in different files allowing any post-processing activity.

Fig. 1. SAT-SURF: view of the case (a) and of the hardware board (b)

TABLE I. SAT-SURF SUB-SYSTEM TECHNICAL SPECIFICATIONS

Parameter	Value
Sensitivity	-160dBm
TTFF	Under 1 second Time-To-First-Fix for Hot and Aided Starts
Augmentation	Supports SBAS: WAAS, EGNOS, MSAS, Assisted GPS
Update rate	4 Hz
Other	Galileo-ready receiver

B. Error correction parameters

The major error contribution in the overall user position accuracy comes from the Ionosphere layer, affecting mostly the vertical component and increasing in such way VDOP (Vertical Dilution of Precision) [8]. The ionospheric parameters taken from the SAT-SURFER log files are ilustrated in Fig.2.

PosID	TOW	WN	Alpha0	Alpha1	Alpha2	Alpha3	Beta0	Beta1	Beta2	Beta3
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
0	318644	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
1	318645	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216
1	318645	1675	1.30E-08	0	-5.96E-08	5.96E-08	110592	-65536	-262144	393216

Fig. 2. Ionospheric correction parameters taken from SAT-SURFER log file

The α and β are the input data of our adaptive positioning algorithm necessary for the mitigation of ionospheric error in the user's position estimation. It will be later shown that we achieve a considerable improvement of the vertical component and a decreased VDOP, after the application of this correction in the main algorithm.

III. GPS SIGNAL ACQUISITION AND TRACKING

The purpose of acquisition is to determine coarse values of carrier frequency and code phase of the satellite signals. Many research works focus on base-band signal processing in the software receivers. [9].

There are several acquisition methods for GPS signals introduced in recent years, which are often implemented in time domain and frequency domain. Among these methods, serial search acquisition is a traditional method for acquisition in CDMA system, but it is time-consuming and performed through hardware in the time domain. In contrast, the conventional parallel in frequency method increases the speed of acquisition by transforming correlation calculation into the frequency domain through DFT (Discrete Fourier Transform) calculation [10-11].

The performance of signal acquisition method was analyzed using the real GPS IF data, which were collected by the SAT-SURF receiver. The GPS receiver was stationary. The intermediate frequency is 4.152 MHz and the sampling frequency is 16.3676 MHz. In our implementation, the conventional parallel in frequency output of the visible satellite with PRN-21 (Pseudo Random Noise code) is shown in Fig. 3.

The quality of the results showed in the Parallel in Frequency (PiF) approach is proportional to the quantity of points used to compute the Fast Fourier Transform (FFT). Another interesting point is the fact that the PiF gives the results in the intermediate frequency range, so in order to get the value of the Doppler it is necessary to subtract the obtained values by the intermediate frequency of the signal, or use other kind of approach to bring down the signal to the base band [11]. After performing the acquisition, control is handed over to the tracking loops, which are used to refine the frequency and code phase parameters. The main purpose of tracking is to refine the carrier frequency and code phase parameters, keep track, and demodulate the navigation data [12]. The values in Table 2 are passed into tracking loop so they can keep track and demodulate the navigation data correctly.

A combination of code tracking loop and carrier tracking loop is used in tracking procedure. In order to extract information from the incoming signals, GPS receivers track them by replicating the PRN code and adjusting its code delay and carrier phase continuously so as to guarantee synchronization with the incoming signal. In Fig.4 a basic code tracking loop is shown. The code tracking loop is to keep track of the code phase of a specific code. The code tracking loop uses a delay lock loop called an early-late tracking loop [13]. Integrate and Dump (I&D) are blocks that accumulate the correlators outputs, and provide their In-phase *I* and Quadrature *Q* components.

As it can be seen in Fig. 5, after 3 steps of the loop the algorithm converges to the correct estimated delay. It is interesting to notice that at this step the Early-Late becomes zero and the Prompt reaches its maximum value.

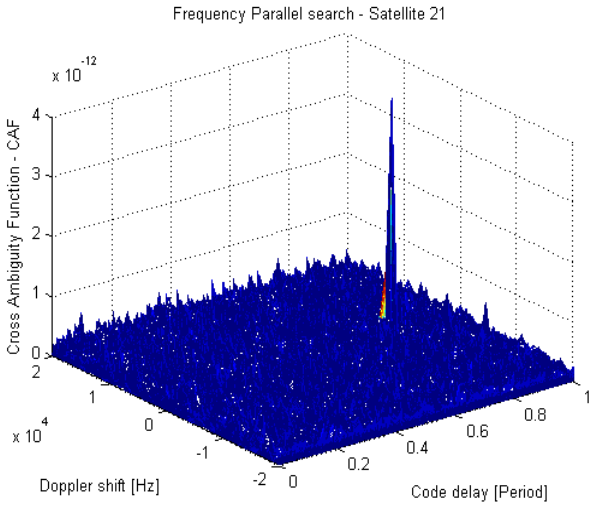


Fig. 3. The correlation results of frequency parallel-search method for PRN-21 satellite

TABLE II. RESULTS FROM PIF GPS SATELLITES IN THE RECEIVED SIGNAL

PRN	Frequency(Hz)	Doppler (Hz)	Code offset
9	4.142e+006	-696.9	15469
12	4.139e+006	1181.8	13756
17	4.158e+006	-2878.8	15194
25	4.150e+006	2636.4	4025
27	4.149e+006	-1303.0	11469
30	4.144e+006	3000.0	15955

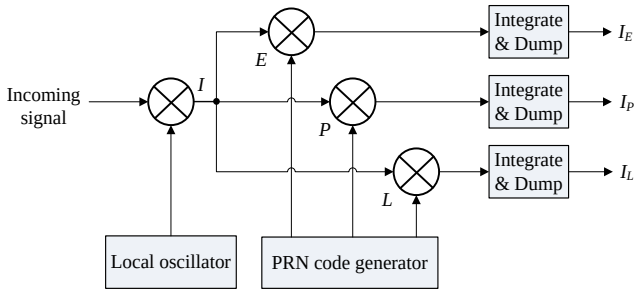


Fig. 4. Basic code tracking loop block diagram

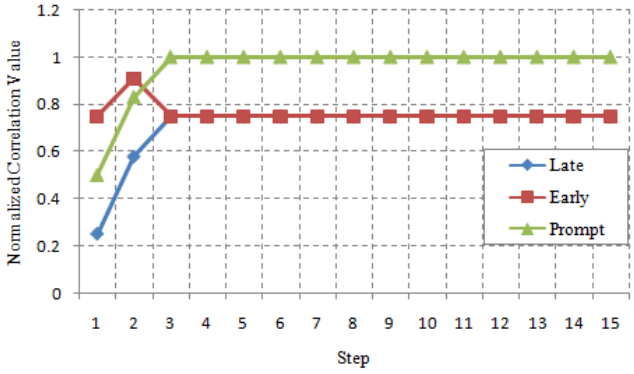


Fig. 5. Modulus of Early, Late and Prompt correlations

IV. ADAPTIVE POSITIONING ALGORITHM IMPLEMENTATION

This section is dedicated to algorithm implementation. We propose an innovative adaptive PVT algorithm compiled in Matlab® environment. The specific computation flow diagram of our positioning algorithm is shown in Fig.6. Initially it is important to extract from the collected data the coordinates of satellites. Since we implement an Iterative Least Squares (ILS) algorithm the method of solving for GPS user's position is to linearize the pseudorange equations and calculate the user position iteratively, starting with a user provided initial position guess [14]. The next step is the calculation of the pseudoranges between satellites and user's position. The algorithm computes the differences between the observed and predicted ranges and gives as output the line-of-sight unit vectors from which it builds the geometry matrix. The convergence of the iterative solution will depend on the geometry of the receiver-satellites system, which, in turn, affects the rank of geometry matrix H. Problems can occur if H is rank-deficient or close to it, which can occur when all the satellites lie in or very close to the same plane in three-dimensional space.

We obtain a least squares optimization only when the solution is over-determined (i.e., number of satellites in view greater than four). When there are only the minimum four measurements, the result is the solution of a set of linear equations. Regardless, solving these equations give us corrections for our initial guess, which can now be reapplied to the initial guess, and the whole process is repeated until the corrections become smaller than a threshold value [15].

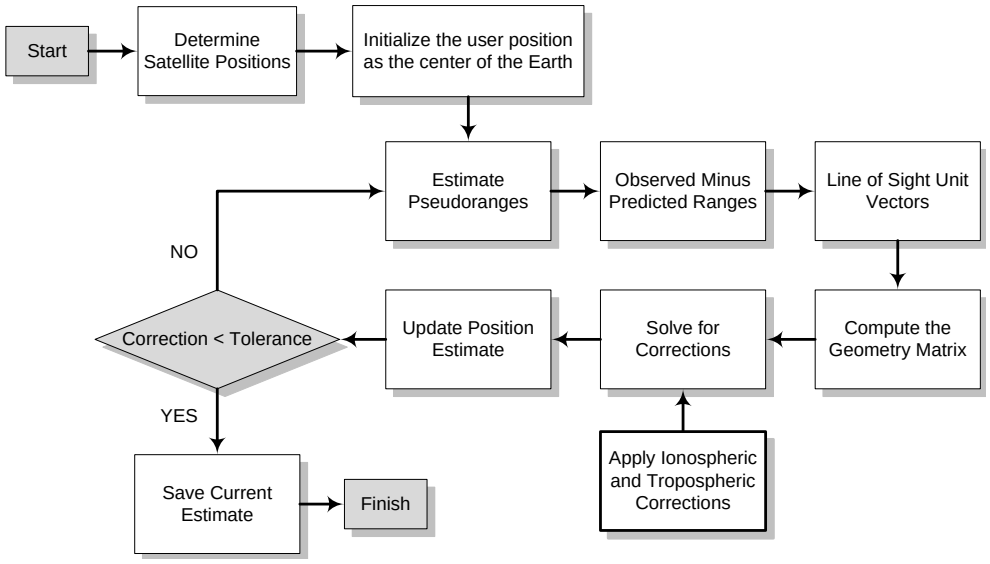


Fig. 6. Computational flow diagram of our positioning algorithm

There are different possibilities of implementing a positioning algorithm [14]. In our approach, data structures are used as a faster and easier way to access the data needed for position computation. In our Matlab® implementation is defined the *"True Position"* only for future comparison of the estimated positions obtained by our positioning algorithm and the true one, in order to graphically depict the precision and accuracy of the estimated positions. In order to evaluate the user position, a linearization scenario is implemented by choosing a linearization point, as our known reference position. Initially the linearization point is set the center of Earth in ECEF (Earth-Centered Earth-Fixed) coordinate system with coordinates *Lp = [0; 0; 0; 0]*. The linearization point will be updated after each TOW (Time of Week) iteration, until in the end of the iteration to become the evaluated user position.

All the parameters that will be used in the algorithm are initially set to zero (initialization process). The chosen linearization point is not a good approximation point because it is very far from our *"True Position"*, but it is suitable for the Cold Start of the receiver (state that the receiver has no information of its position).

During our measurements we collected a large amount of data for a total of 2340 TOW-s. The positioning algorithm is tested for different number of iterations and the results obtained for the user position were approximately the same. This is due to the long observation time and due to the fact that the minimum number of fixed satellites were 6 (enough to properly estimate user's position). In Matlab® environment, simulation time is not a crucial issue but in real receivers, time is a very important constraint.

The Navigation Solution in a first-order approximation is given by the following code lines:

```
SatP(i,:) = [xs ys zs];
rho_hat(i) = norm(SatP(i,:) - Lp);
a(i,:) = (SatP(i,:) - Lp)/rho_hat(i);
```

The first one addresses the satellite coordinate's triplet (*Xs, Ys, Zs*) , which are used in the second line in order to evaluate the geometrical distance between the satellite position and the linearization point *Lp*. After this process the *a* coefficients of the geometric matrix are written and it is important to say that for the first iteration and for the first TOW, we assume that the satellite clock and user clock are synchronized. This happens only for the first TOW, because the coordinates of the updated linearization point will be used as input for the successive TOW.

Since the strength quality of the signal is defined by the Carrier-to-Noise density Ratio, which is the ratio of the power level of the signal in 1 Hz bandwidth, it is important to properly weight satellites with low values of C/N_0. Elevation angles and C/N_0 values, as recorded by the receiver, are used to model the pseudorange observations noise variance. The choice of the weight matrix is optimal when it equals the inverse of the variance-covariance matrix of the observations [16]. We implemented the model in [17] which uses the C/N_0 values of the GPS signals to estimate weights for least square adjustment. Using this approach we achieved an improvement on the position estimation mostly in the vertical component.

V. Ionospheric and Tropospheric Correction Models

The focus of this section is to evaluate the ionospheric and tropospheric effect on GPS positioning solution. The pseudoranges are affected by errors, which can be modeled as Gaussian random variables, with zero mean, independent and

identically distributed, with variance σ^2_{UERE} [15]. The errors affecting the pseudoranges can be expressed by (1).

$$\rho = \sqrt{(x_i - x_{Lp})^2 + (y_i - y_{Lp})^2 + (z_i - z_{Lp})^2} - c \cdot t_{Lp} + c \cdot T_a \quad (1)$$

Where $T_a = T_{Iono} + T_{Trop}$ is the sum of Ionospheric and Tropospheric error contributions, respectively. These two types of corrections are described in details in the following subsections.

A. Ionospheric Corrections

Ionospheric corrections are implemented based on the Klobuchar model [3] which uses as input the parameters shown in Table 3. We designed a function *ionogen.m* to calculate the delay caused by Ionosphere layer, which was called in our main PVT algorithm. Two are the main inputs of the ionospheric correction function. The first one is *PER* which is the period of the cosine function and implicates the interval of the ionospheric activity in daytime. It is expressed by (2), whose inputs are taken from the ionosphere log file.

$$PER = \beta_0 + \beta_1 \cdot lat_m + \beta_2 \cdot lat_m^2 + \beta_3 \cdot lat_m^3 \quad (2)$$

Where lat_m is the geomagnetic latitude of the Earth's projection of the ionospheric intersection point (mean ionospheric height assumed to be 350 km). The second input is the amplitude of the model given by (3).

$$AMP = \alpha_0 + \alpha_1 \cdot lat_m + \alpha_2 \cdot lat_m^2 + \alpha_3 \cdot lat_m^3 \quad (3)$$

The inputs of the Klobuchar model were taken by loading the Elevation and Azimuth angles for each TOW and number of fixed satellites. We observed that these coefficients are constant even for different TOW (Fig. 2) and this result is due to the fact that ionospheric parameters do not change in a short measurement time.

B. Tropospheric Corrections

The signal refraction in the troposphere is separated into two components: the dry and the wet component, where the dry component contributes about 90 % of the total tropospheric delay. The tropospheric delay is approximated by using the Hopfield model [6], whose inputs in our algorithm are:

- T - *Temperature* in °C.
- P - *Pressure* in hPa.
- H_u - *humidity ratio* in %.
- R - *Earth radius:* R = 6371 km.
- θ_e - *Satellite Elevation angle.*

This model is based on the relationship between the dry refractivity at height *h* to the surface of Earth. We designed a function in Matlab® named *tropogen.m* to calculate the delay caused by the Troposphere layer, as a function of elevation angle represented by the following equations

$$\Delta\rho_{Trop}(\theta_e) = \Delta\rho_{dry}(\theta_e) + \Delta\rho_{wet}(\theta_e) \quad (4)$$

Equation (4) represents the total Tropospheric error contribution where $\Delta\rho_{wet} = K_w[I(h_w) - b]$ and $\Delta\rho_{dry} = K_d[I(h_d) - b]$. The humidity ratio in % in dry and wet conditions is given by (5):

$$H_w = 11000 \text{ and } H_d = 40136 + (148.72 \cdot T) \quad (5)$$

TABLE III. INPUT PARAMETERS OF KLOBUCHAR CORRECTION MODEL

Receiver generated terms	
λ_u	User Geodetic Latitude WGS 84 (semi – circles)
φ_u	User Geodetic Longitude WGS 84 (semi – circles)
E	Elevation angle between the user and the satellite, measure clockwise positive from the true north (semi- circles)
A	Geodetic azimuth angle of the satellite
GPS time	Receiver's computed system time
Satellite transmitted terms	
α_n	Coefficients of a cubic equation representing the amplitude of the delay
β_n	Coefficients of a cubic equation representing the period (*PER)* of the model

VI. ANALYSIS OF THE RESULTS

In this section are shown the results of our work. The reference frame used is the ECEF Cartesian coordinate system. In Fig. 7 the time evolution of Geometrical Dilution Of Precision (GDOP) and the number of satellites are shown. We observe in table 4 that for all the TOW-s taken into consideration, the minimum number of fixed satellites is six which is enough to properly estimate the user position because are required at least four satellites. When the number of fixed satellites decreases, we observe increased values of GDOP, for instance when the number of fixed satellites goes from 13 to 6 the value of GDOP is increased from 1.59 to 5.29. When the number of fixed satellites increases, so more satellites come in view, the proper values of GDOP decrease because a better estimation of the receiver's position is achieved.

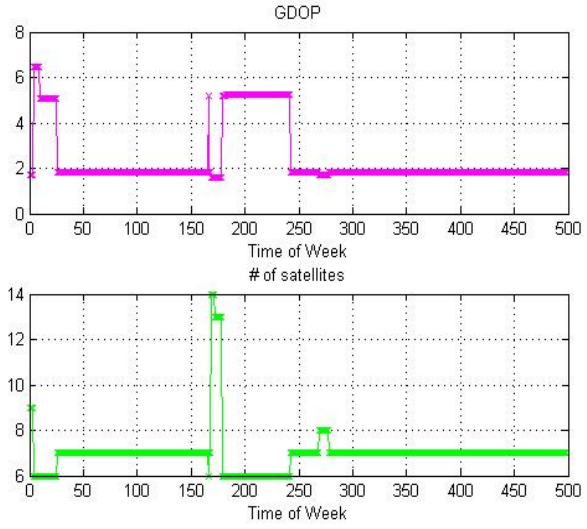


Fig. 7. The change of GDOP values and of number of satellites over TOW

TABLE IV. GDOP VALUES AND NUMBER OF SATELLITES IN VIEW FOR ALL TOW

Value	Minimum	Mean	Maximum
GDOP [m]	1.59	2.422	6.49
Number of Satellites	6	7	14

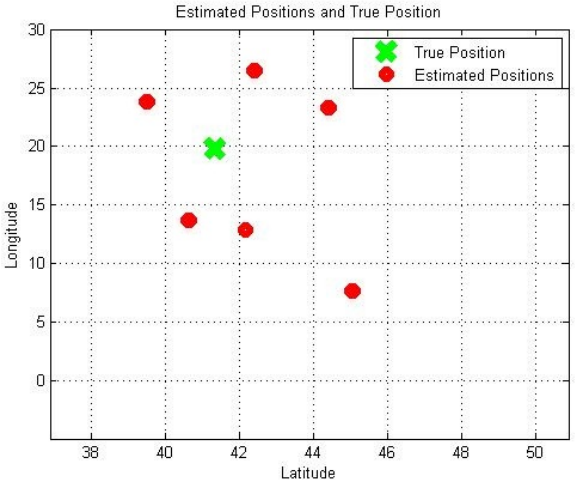


Fig. 8. The true and estimated position in Geographic coordinates

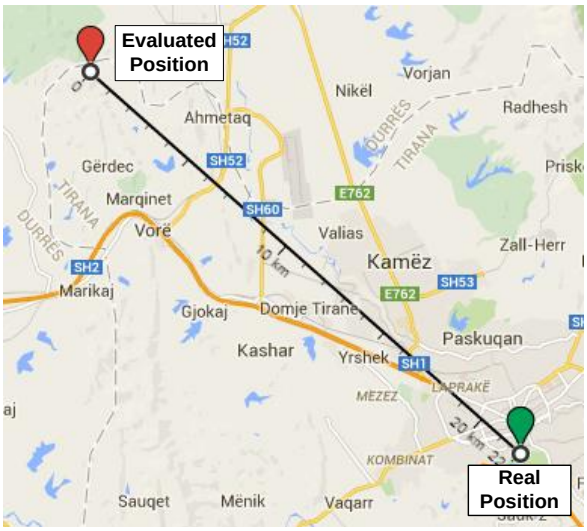


Fig. 9. Estimated position for the first 10 Times Of Week

In Fig. 8 are plotted the true position of the receiver and the cloud of points which represents the estimated position as an output of the positioning algorithm.

After running the positioning algorithm with the raw data of the first 10 Times Of Week the obtained estimated position (Fig. 9) has the following coordinates: Latitude = 41.454^0 and Longitude = 19.626^0 which is far from the true position. This is due to the linearization point which at the beginning is *Lp* and to the fact that the receiver is set up for the first time (cold start). The receiver in cold start mode has no clue where its position might be and the first linearization point is far from the true position.

Fig. 10. Estimated user's position from the adaptive PVT algorithm

TABLE V. SUMMARY OF DIFFERENT TRIALS COMPUTED FOR THE PVT SOLUTION

User's Position	Latitude	Longitude	Height (m)
Without correction	41.3165^0	19.8215^0	61.134
Ionosphere correction	41.3165^0	19.8215^0	18.521
Troposphere correction	41.3165^0	19.8215^0	59.231
Iono + Tropo correction	41.3165^0	19.8215^0	14.275

In Fig. 10 is shown the estimated position of the user after the iterations for all Times of Week and it has these coordinates: Latitude=41.3165^0 and Longitude=19.8215^0 which are close to the true position (Latitude=41.3169^0 and Longitude=19.8215^0). The satellite-user geometry can have a large impact on the accuracy of the PVT estimates obtained from GPS. In other words, some satellite-user geometries will result in a higher accuracy solution than others. As such, it is useful to have a way of comparing different satellite-user geometries. The metric normally used for measuring this impact is dilution of precision (DOP), which represents the degree to which satellite-user geometry dilutes the accuracy of the PVT. DOPs can be viewed as the link between the pseudorange errors and PVT estimation errors. Since DOPs change as the user-satellite geometry changes over time as illustrated in Fig.7, this implies that a given level of pseudorange measurement error will translate into different levels of PVT errors.

After applying the ionospheric and tropospheric correction models, the error in the vertical component (height z) is significantly reduced. Figure 8 shows the estimated positions and the true position in Geographical coordinates for a better understanding of the atmospheric residual errors. The Klobuchar model reduces the vertical error with a value equal to 42.6 m. The Tropospheric Hopfield model applied in our adaptive PVT algorithm, gives a slight correction to the vertical error in the amount of 1.9 m. This was an expected outcome because Tropospheric error's impact is lower compared to the Ionospheric one, in the total error contribution. These important results are summarized in Table 5.

Finally, the user's position estimated by our adaptive ILS positioning algorithm for all GPS epochs or TOWs is: [41.3165^0 *N*; 19.8215^0 *E*; 14.275 m]. This estimated position is very close to the true position which is illustrated in Fig. 10.

VII. Conclusion and Future Work

The aim of this paper was to evaluate the positioning performance of a single-frequency software GPS receiver using Ionosperic and Tropospheric corrections. We proposed an adaptive ILS algorithm, where we integrated Klobuchar and Hopfield mathematical correction models, enabling data post-processing. In our measurement process we used the SAT-SURF receiver. In order to minimize the impact of large errors in the position estimation, we applied the Weighted Matrix. In the first ten TOW-s test we obtained very bad results in the user position estimation, this was due to cold start of the receiver (where the receiver has no clue about its position) and because the initial linearization point was chosen to be very far from the user's True Position. Since the goal of a positioning algorithm is to provide the user position in a minimum number of iterations we show that three iterations were enough to fulfill this requirement. The final user position obtained by our positioning algorithm was 41.3165^0 *North*, 19.8215^0 *East*, 61.134 m *Up*. Applying the Klobuchar model for Ionospheric correction, a reduction by 42.6 m of the vertical error was achieved; however this model did not affect significantly the horizontal positioning. On the other hand, the integration of Hopfield Tropospheric model in our positioning algorithm, gave a slight improvement of the vertical error by 4.25 m compared to ionospheric correction. This is a good result, taking into account that our receiver is a mass market receiver working in single frequency. In our future work we will focus on the mitigation of other error's contribution such as relativistic, ephemerides and satellite clock errors. We will also investigate the positioning performance achieved after the application of EGNOS and differential corrections, using double frequency GPS receivers for Precise Point Positioning applications.

References

[1] R. Warnant, K. Ivan, P. Marinov, M. Bavier, and S. Lejeune, "Ionospheric and geomagnetic conditions during periods of degraded GPS position accuracy: 2.RTK events during disturbed and quiet geomagnetic conditions", Advances in Space Research,Vol. 39, No. 5., pp.881-888, 2007.

[2] G . Hochegger, B. Nava, S.M. Radicella and R. Leitinger "A family of ionospheric models for different uses", Phys. Chem. Earth, 25 (4), 307-310, 2000.

[3] J.A. Klobuchar, "Ionospheric time-delay algorithm for single-frequency GPS users", IEEE Trans. Aerosp. Electron. Syst., AES-23 (3), 325-331, 1987.

[4] B. Hofmann-Wellenhof, H. Lichtenegger, J. Collins, "Global Positioning System:Theory and Practice", 5th revised edition, Springer-Verlag, 382 pp. 2001.

[5] R.B. Langley, "Propagation of the GPS Signals", In: Kleusberg, A. and Teunissen, P.J.G. (eds), GPS for Geodesy (2nd edition), Springer-Verlag, Berlin Heidelberg New York, 111-150, 1998.

[6] H.S. Hopfield, "Two–quadratic tropospheric refractivity profile for correction satellite data", Journal of Geophysical Research, 74(18), 4487 – 4499, 1969.

[7] P. X. Quang, L. Lo Presti, F. Dominici, G. Marucco, "SAT-SURF and SAT-SURFER: a flexible platform for both R&D and training on GNSS". In: 2nd GNSS Vulnerabilities and Solutions 2009 Conference, Baska, Krk Island, Croatia, 2-5 September, 2009. pp. 1-12.

[8] J.L. Leva, "Relationship between navigation vertical error, VDOP, and pseudo-range error in GPS", Aerospace and Electronic Systems, IEEE Transactions, vol.30, no.4, pp.1138,1142, Oct 1994.

[9] L. Dong, Ch. Ma, and G. Lachapelle, "Implementation and Verification of a Software-Based IF GPS Signal Simulator," Proceedings of the 2004 National Technical Meeting of The Institute of Navigation, San Diego, CA, January 2004, pp. 378-389.

[10] J. Tian and L. Yang, "A Novel GNSS Weak Signal Acquisition Using Wavelet Denoising Method," Proceedings of the 2008 National Technical Meeting of The Institute of Navigation, San Diego, CA, January 2008, pp. 303-309.

[11] B. Y. T. James, "Fundamentals of global positioning system receivers a software approach", *A John Wiley&sons*, New York, 2004.

[12] P. Lian, G. Lachapelle, and Ch. Ma, "Improving Tracking Performance of PLL in High Dynamics Applications," Proceedings of the 2005 National Technical Meeting of The Institute of Navigation, San Diego, CA, January 2005, pp. 1042-1052.

[13] R. Peter and B. Nicolaj, "Design of a single frequency GPS software receiver", *Aalborg University*, pp. 31-35, 2004.

[14] W. Li, Z. Yuan, B. Chen, and W. Zhao, "Performance comparison of positioning algorithms for complex GPS systems", Distributed Computing Systems Workshops, 32nd International Conference on Distributed Computing Systems (ICDCSW), pp.273-278, Macau, China,18-21 June 2012.

[15] P. Misra, and P. Enge, "Global Positioning System: Signals, Measurements and Performance, Revised Second Edition", Lincoln, MA: Ganga-Jamuna Press, 2011, ISBN 0-9709544-1-7.

[16] C.C.J.M. Tiberius, "The GPS data weight matrix: what are the issues?", Proceedings of the 1999 National Technical Meeting of The Institute of Navigation, January 25 - 27, 1999, Catamaran Resort Hotel, San Diego, CA, pp. 219 – 227.

[17] H. Hartinger and F. Brunner, "Variances of gps phase observations :The sigma-e model," GPS Solutions, vol. 2, pp. 35–43, 1999.

Author Profile

Alban RAKIPI has graduated the Faculty of Information Technology, Polytechnic University of Tirana in 2009. He holds a Master of Sciences diploma in Telecommunication Engineering from 2011 and a second level specializing master diploma in "Navigation and Related Applications" from Polytechnic University of Turin, Italy. Currently he is a full-time Lecturer within the Department of Electronics and Telecommunications at Faculty of Information Technology, Polytechnic University of Tirana.

His work focuses on signal processing, satellite positioning systems and technologies, GNSS applications, integrity monitoring, mobile ad hoc routing algorithms and network protocols.

Steganography: Applying and Evaluating Two Algorithms for Embedding Audio Data in an Image

Khaled Nasser ElSayed
Computer Science Department, Umm Al-Qura University

***Abstract*—Information transmission is increasing with grow of using WEB. So, information security has become very important. Security of data and information is the major task for scientists and political and military people. One of the most secure methods is embedding data (steganography) in different media like text, audio, digital images. This paper presents two experiments in steganography of digital audio data file. It applies empirically, two algorithms in steganography in images through random insertion of digital audio data using bytes and pixels in image files. Finally, it evaluates both experiments, in order to enhance security of transmitted data.**

Keywords—Steganography; Encryption and Decryption; Data and Information Security; Data Hiding; Images; Data Communication

I. Introduction

Nowadays, data and information have become the most important hot issue. Technology of transmission and communication of data and information between sites in the same country or overseas are done through LANs, WANs, or WEB networks. This process is done through leased lines, microwave, or satellites. From academic wise, information security is the science that cares and searches in the theories and strategies of providing information protection against violation of unauthorized peoples. From technology wise, information security is tools and procedures needed to ensure (grant) information protection from inside and outside dangers. From law wise, information security is the process and studies performed to protect the security and integrity of data and information against its violations.

Data hiding or embedding refers to the nearly invisible embedding of information within a host data set such as text, image, or video [1], [2]. In steganographic applications, the hidden data are a secret message whose mere presence within the host data set should be undetectable; a classical example is that of a prisoner communicating with the outside world under the supervision of a prison warden. In this context, the data hiding represents a useful alternative to the construction of a hypermedia document, which may be less convenient to manipulate [3].

Steganography is the art and science of hiding information by embedding messages within other, seemingly harmless message. Steganography means "covering writing" in Greek. As the goal of steganography is to hide the presence of a message and to create a covert channel, it can be seen as the complement of cryptography whose goal is to hide the content of the message [4].

Next, section II, will emphasizes on related research work, while, section III, will present the techniques used in information security, like cryptography and steganography. Section IV will emphasize steganography methods, while section V and VI will present and evaluate two experiments.

II. Related Work

In recent works [5][6][7], it has been shown that digital data can be effectively hidden in an image so as to satisfy the criteria that the degradation to the host image is imperceptible and it should be possible to recover the hidden under a variety of attack. The main idea is to view the data hiding problem as a communication with channel side information [8] [9].

Steganography can be used in a lot of useful applications. For example copyright control of materials, to enhance the robustness of an image search engines and smart identity cards where the details of individuals are embedded in their photographs [10]. Steganography may be classified as pure, symmetric and asymmetric. While pure steganography does not need any exchange of information, symmetric and asymmetric need to exchange of keys prior sending the messages. Steganography is highly dependent on the type of media being used to hide the information. Medium being commonly used include text, images, audio files, and network protocols used in network transmissions [11].

Chandramouli and Memon [12] developed the most common method used to hide the message which involved the usage of Least Significant Bit (LSB). They apply the filtering masking and transformation on the cover media.

Abdullatif and Shukur [13] proposed a blind color image steganography method that embeds secret message by spraying theme on the blocks in the high order bits in color channel such as blue. However it also depends on the constant sequence spread spectrum method to survive loss compression image like JPG.

Atawneh and et al [14] presented common approaches and tools that are used in digital image steganography. It is shown mathematically and graphically. The differences between steganography, cryptography and watermarking technique are discussed. The authors also highlighted the current steganography tools and demonstrate how the secret information is embedded into image through the tools.

Ameen and et al [15] presented two methods for destroying steganography content in an image that are the overwriting and the de-noising method. The overwriting method is a random data that can be written again over steganographic images

while the de-noising method uses two kinds of destruction techniques that are filtering and discrete wavelet techniques. These two methods have been simulated and evaluated over two types of hiding techniques that are Least Significant Bit LSB technique and Discrete Cosine Transform DCT technique.

Hamid and et al. [16] presented the use of an image file as a carrier and the taxonomy of current steganographic techniques. The authors analyzed and discussed steganography techniques for their ability in information hiding and the robustness to different image processing attacks. They also briefly discussed steganoanalysis which is the science of attacking steganography.

The proposed work emphasizes on information protection against unauthorized persons while passing through networks. It presents cryptography and steganography algorithms. Then it presents the process of hiding of a message (digital audio data file) in an image file (cover images) using random insertion techniques through applying two experiments: insertion using byte level and insertion using pixel level. Finally, it gives evaluation for both applied algorithms.

III. Information Security Techniques

Our problem is distinguishing between important and the most important information, and, thus protecting information against violation. Data and information are used by all, individuals, companies, organizations, and countries.

Information security techniques are the procedures, tools, and products used to protect or at least decrease danger and violation of information, networks, information systems and their databases. There are many security tools already have been used in information environment like identity-passwords and fire walls, cryptography, intrusion detections, and anti-virus systems.

A. Cryptography

Cryptography is the transformation of data and information to unclear and non- understood code (looks has no means) to prevent unauthorized access of information. While, decryption is getting (extracting) the original information from the encrypted one. In the time being, cryptography gets more attention in information security field. This is because cryptography is the most important security techniques to provide secretly, integrity, and availability of information. In general, cryptography, and its application, specially, electronic signature, is the only way for grantee the responsibility over electronic nets.

Nowadays, internet is largest multimedia for information transmission. Keys are used in data encryption and decryption, and are based on complicated mathematical formulas (algorithms).

1) Symmetric cryptography (secret key): Where, both sender and receiver use the same secret key in message encryption and decryption they agree on using a pass phrase. The pass phrase can use capital, small, and other characters. The cryptography software transfers the pass phrase to binary number and adding other symbols to increase its length. The resulted binary number constitutes the cryptography key of the message. After receiving the encrypted message, the receiver uses the same phrase to retrieve the original message. The problem of this type is the unsafe distribution of the secret key.

2) Un-Symmetric cryptography (general key): comes due to the unsafe distribution of key. It uses two related key instead of one key; public key and private key. The private key is known only by the sender and used to encryption and decryption of the message. While, the general key is known by multiple user and used in decryption to retrieve the original message that was encrypted using the private key. The owner of the private key can retrieve the message using the general key. Although, this message is better than the symmetric one, it is not away from violation.

B. Steganography

Steganography can be done through embedding or inserting (hiding) messages in text, voice, or image file. To perform that correctly, data integrity should be the same after applying steganography. Data can't be protected by making it single block, it should be fragmented into several blocks during execution. These blocks should be secured against modification through attack. Also, we should predict that those blocks could be distorted so symbol corrections should be used.

1) Steganography in text: Where there are three theories. The first theory is, Line-Shift Coding, which code text lines vertically. The problem of this theory is that most of text symbol coding is visible to the reader and pixels between texts could be measured manually or automatically. The second theory is, Word-Shift Coding, which depends on coding document through moving horizontal positions of words of a text. This method is less visibly to the reader. The third theory is, Feature Coding, which depends on random distribution of text which makes violating is too difficult.

2) Steganography in an audio: Where too factors should be considered: Digital representation of audio and signal transmission. Digital representation has two major features : simple quantization method, where quality degree of digital audio is represented in 16 bits by Windows Audio-Visual(WAV) and Audio Interchange File Format(AIFF), and temporal sampling rate, which is in some ranges. Two theories are used: (1) Low-bit Encoding and (2) Phase Coding.

There is four media for signal transmission: (1) Digital End-to-end Environment: if audio file is copied directly with no change from machine to another, it will be sent through this environment. (2)Increased/Decreased Re-sampling Environment: signal is re-encoded to the lowest or the highest coding rate. This environment is suitable for steganography. (3)Analog Transmission and Re-sampling: this environment is used when signal is transformed to analog system. (4) Over Air Sampling: this environment when signal is over air from microphone and is loaded on media for transmission.

3) Steganography in an image: Which inserts the data file (hided message) inside the image file (cover image)? The message could be in normal text or encrypted text or even another image. Fig. 1, presents the general scheme of

steganography in an image at the sender and the receiver. The problem now is how to insert the message in the cover image. Next section will emphasizes these methods.

IV. Steganography Methods

There are some methods used to hide information in digital images could be applied on different image files with different level of success.

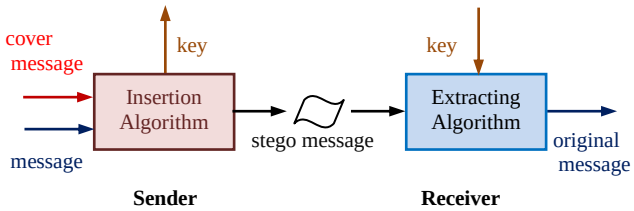


Fig. 1. General scheme of Steganography in images

A. Least Significant bit insertions

Embedding information in an image file by insertion in least significant bits is a famous and simple method, but it is sensitive for any manipulation of images, even it is was simple.

To insert information in least significant bit of an image bytes encoded in 24 bits, 3 bits are stored in each image pixel. For example, in an image of 768×1024 pixels, we can insert 2,359,295 digits of information (294,912 bytes or characters). If the inserted information was compressed before insertion, more characters could be inserted. The stego image that includes the inserted data will look as it will before insertion. We can insert information in the least significant bit and the following bit, without any change detected in the image viewing.

As example, to insert the character A (10000011) in the following 3 pixels (assuming no compression) (9 bytes) :

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

After insertion, the resulted 3 pixels will be as follow :

(00100111 11101000 11001000)
(00100110 11001000 11101000)
(11001001 00100111 11101001)

B. Algorithms and Transformation

Steganography using least significant bits is an easy and fast method, but it is too sensitive for any little change done in the image due to manipulation, processing, or lossless compression.

JPEG-JSTEG is a steganography method that integrates compressing algorithm with information steganography. It generates stego image by JPEG methodology from inputs of lossless cover image and message to be hided. JPEG software is modified to accept one digit steganography and the output file is TFIF standard. The TFIF standard consists of lossless and lossless parts. Software contains the message and the cover image by JPEG algorithm to generate steganography image in lossless JPEG.

JPEG image uses discrete cosine transform (DCT) to perform compression. This transform is lossless compression, because computing of cosine value exactly is not possible, and replicated computation that uses low precession numbers results in circulation errors in the final results. Difference ranges between the original given values and the extracted values depend on the applied method in transformation computation (DCT).

Beside DCT, fast Fourier Transform could be used to manipulate images and wavelet transform. This method maintains image in quality degree higher than the tools that depends only on a least significant bits. At using extra coding for shapes, we should compensate between message size and insertion. If the message has a small size, it could be inserted many times, but large size message could be inserted once, because of the huge part of the image occupied by the message.

V. Experiment 1: Random Insertion Using Bytes

This method transforms both image and message into two arrays of bytes, then it generates random positions in bytes range of image, where number of positions is equal to number of message bytes or characters (when size of message array is less than or equal to image array). Then, message bytes are inserted in the positions randomly generated in image bytes. Finally, those positions are kept in a key array.

A. Insertion Algorithm

Fig. 2, presents the algorithm used in random insertion using bytes. Assume that the bytes array shown in Fig. 3-a, resulted from transforming certain message into bytes. Also, I want to insert that message bytes array in the image bytes array shown in Fig. 3-b. So, I used a random function that generates random numbers (like those listed in Fig. 3-c).

This list of numbers represented in the key array, will be the numbers of positions in the image bytes array, where the message bytes will be inserted. Also, their number is equal to the message bytes array size. Those positions will be stored in the key array. The result of inserting the message bytes array (3-a) in the random positions (3-c) of image bytes array (3-b) is shown in Fig. 3-d (bold bytes represents the inserted bytes).

B. Results and Evaluation

Steganography in images using bytes algorithm emphasized the following results:

- Random generating lets discovering insertion positions more difficult than other steganography methods.
- Distortion looks as hashed points in the image, and it can't depend on image properties like brightness or sharpness or others.
- Distortion is too much because message insertion is done randomly not selectively for less important positions as in least significant method.
- Distortion could be reduced by reducing inserted digits in image bytes, as example, by inserting half a byte from message bytes in one image byte (or more or less).

- When the message is of huge size and the image is somehow of a small size, this method is not applicable.

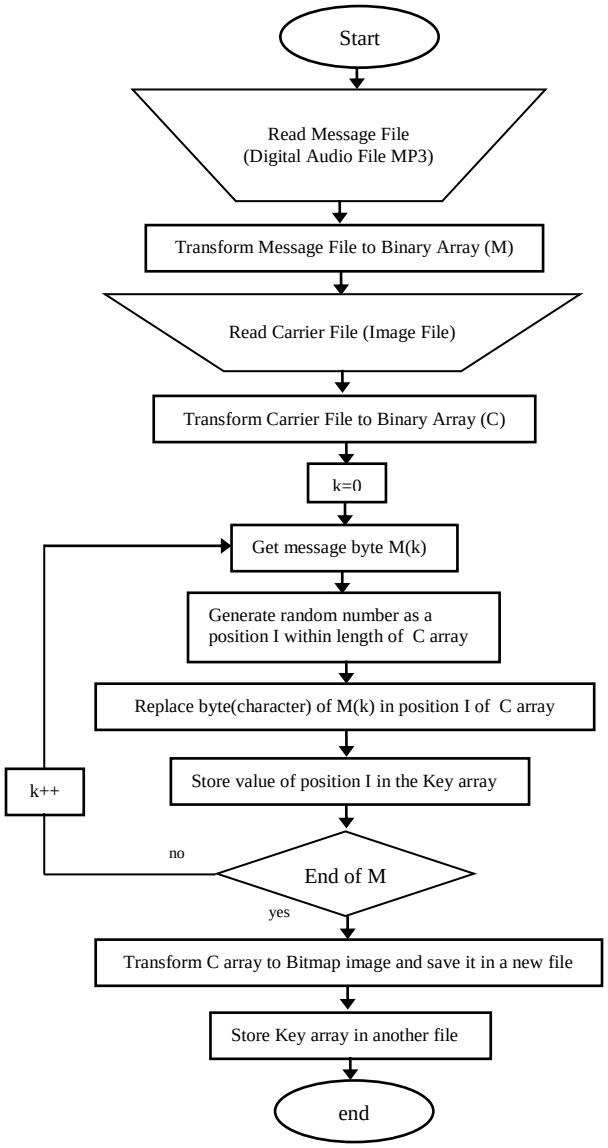


Fig. 2. Algorithm of steganography in images using bytes

a. message bytes array

1	00010101
2	01011010
3	11101001
4	00100101
5	00101010
6	01010010

b. image bytes array

1	00001111	11	11111111	21	00000111
2	00000111	12	00001111	22	11000000
3	11000011	13	11100000	23	00000111
4	11111100	14	11111111	24	11011000
5	00111111	15	11110000	25	00000000
6	00011111	16	00000111	26	00011111
7	11111101	17	11110011	27	11110001
8	01110111	18	11100010	28	11111000
9	11111110	19	11111100	29	00111111
10	00111111	20	01111111	30	00000000

c. position randomly generated

(8, 3, 19, 20, 12, 28)

d. resulted stego image bytes array

1	00001111	11	11111111	21	00000111
2	00000111	12	**00101010**	22	11000000
3	**01011010**	13	11100000	23	00000111
4	11111100	14	11111111	24	11011000
5	00111111	15	11110000	25	00000000
6	00011111	16	00000111	26	00011111
7	11111101	17	11110011	27	11110001
8	**00010101**	18	11100010	28	**01010010**
9	11111110	19	**11101001**	29	00111111
10	00111111	20	**00100101**	30	00000000

Fig. 3. steps and results of steganography using bytes(continued)

VI. Experiment 2: Random Insertion Using Pixels

In the method of random insertion using pixels, image is transformed to pixels array, each pixel is represented in 3 bytes, and message is transformed to a byte array. Then, random positions are generated in image pixels rang, with number of positions equal to message bytes (when the size of message bytes array is less or equal to the size of image pixels array). Each message byte is inserted in a position of image pixels, where positions are generated randomly and kept in the key array.

A. Insertion of a Byte in a Pixel

Assume that we have the byte (00101010) shown in Fig. 4-a, and we want to insert it in a certain pixel, shown in Fig. 4-b. The resulted pixels after insertion are shown in Fig. 4-c.

a. the message byte to be inserted.

(00101010)

b. the original pixels of an image.

(01010100	10101110	10111111)
R	G	B
84	174	191

c. pixels of image after insertion

(01010**001**	10101**010**	10111**10**)
R	G	B
81	170	190

Fig. 4. Inserting a byte in certain pixel

We can notice that last 3 bits in the first two bytes and the last 2 bits in the third byte contain message byte value that will result in simple change rate in each color of pixels colors. These bits are the least significant bits. By this method, the random insertion using pixels will be done. Distortion could be decreased if we insert each message byte in three pixels. We can see Fig. 5, which presents insertion of a message byte in 3 pixels.

Notice the change occurs only in the first digit in the first 8 bytes. It is clear that the distortion rate at inserting 1 byte in 3 pixels will be less than the distortion happen at inserting 1 byte in 1 pixel. But, the size of a message to be inserted in certain image should be of smaller size. We can use a moderate solution by inserting 2 message bytes in 3 pixels.

a. the message byte to be inserted.

(00101010)

b. the original pixels of an image.

01000101	10001111	10110010
01010111	11101110	10111001
01010000	10111110	00001111

c. pixels of the image after insertion

0100010**0**	1000111**0**	1011001**1**
0101011**0**	1110111**1**	1011100**0**
0101000**1**	1011111**0**	00001111

Fig. 5. Inserting a byte in certain pixel

B. Random Insertion Algorithm

Assume that the bytes array shown in Fig. 6-a, resulted from transforming certain message into the byte. And, we want to insert that message bytes array in the image pixels shown in Fig. 6-b.

To insert the message bytes array listed in Fig. 6-a in the image pixels in Fig. 6-b, we use a random function that generates random numbers (like those listed in Fig. 6-c). This list of numbers represents the key array, that will be the positions numbers in the image bytes array, where the message bytes will be inserted, and their number is equal to the message bytes array size. Those positions will be stored in the key array. The result of inserting the message bytes array (6-a) in the random positions (6-c) of image pixels (6-b) is shown in Fig. 6-d (bold digits represents the inserted bytes). Fig. 7, presents the algorithm used in steganography in image using pixels.

a. message bytes array

1	00010101
2	01011010
3	11101001

b. image bytes array

1	00001111	11111111	00000111
2	00000111	00001111	11000000
3	11000011	11100000	00000111
4	11111100	11111111	11011000
5	00111111	11110000	00000000
6	00011111	00000111	00011111
7	11111101	11110011	11110001
8	01110111	11100010	11111000
9	11111110	11111100	00111111
10	00111111	01111111	00000000

c. position randomly generated

(3,6,10)

d. resulted stego image bytes array

1	00001111	11111111	00000111
2	00000111	00001111	11000000
3	11000**000**	11100**101**	00000**01**
4	11111100	11111111	11011000
5	00111111	11110000	00000000
6	00011**010**	00000**110**	00011**10**
7	11111101	11110011	11110001
8	01110111	11100010	11111000
9	11111110	11111100	00111111
10	00111**111**	01111**010**	00000**01**

Fig. 6. Steps and results of steganography using pixels (continued)

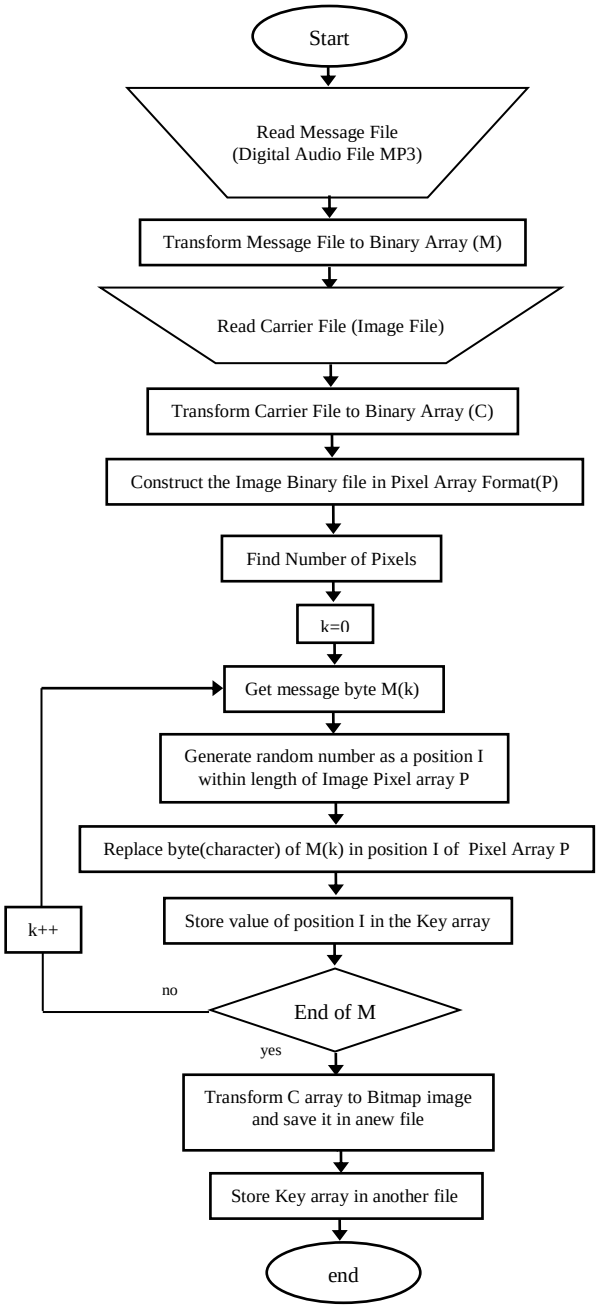


Fig. 7. Algorithm of steganography in images using pixels

C. Results and Evaluation

Steganography in images using pixels algorithm emphasized the following results:

- Distortion will be very small or even null, because of using least significant bits.
- Random generation of hiding position make discovering them is very difficult in contrast with other methods of steganography, where message could be discovered once the message used is known.

VII. Conclusions

This paper presented two algorithms in steganography in images through random insertion (hiding) of data using bytes and pixels. Newly, generating function of random values were built specially for steganography. It was impossible to find out message data, in contrast with other methods. Random generating lets discovering insertion positions more difficult than other steganography methods.

In the method of random insertion using bytes, both image and message were converted into two arrays of bytes, then it generated random positions in bytes range of image, where number of positions is equal to number of message bytes or characters. Then, message bytes were inserted in the positions randomly generated in image bytes. Finally, those positions were kept in a key array. Distortion was too much because message insertion is done randomly not selectively for less important positions as in least significant method. Distortion could be reduced by reducing inserted digits in image bytes, as example, by inserting half a byte from message bytes in one image byte (or more or less).

While, in the method of random insertion using pixels, image was transformed to pixels array, each pixel was represented in 3 bytes, and message was transformed to byte array. Then, random positions were generated in image pixels rang with number of positions equal to message bytes. Each message byte was inserted in a position of image pixels, where positions were generated randomly and kept in the key array. steganography in images using pixels algorithm emphasized that distortion would be very small or even null, because of using least significant bits.

Those experiments were applied to improve data transmission security. So, future work will evaluate some different insertion techniques and evaluate using different cover messages, in order to minimize distortion in the hided messages.

Refrences

[1] W. Bender, D. Gruhl and N. Morimoto, "Techniques for Data Hiding", IBM System Journal, Vol. 35, 1996.

[2] M. D. Swanson, M. Kobayashi and A. H. Tewfik, "Multimedia Data-Embedding and Watermarking Strategies", Proc. IEEE, Vol. 86, No. 12, pp. 1064-1087, June 1986.

[3] P. Moulin and M. Kivanc Mihcak, "A Framework for Evaluating the Data-Hiding Capacity of Image Sources", IEEE Int. Conf. On Image Processing, Vancouver, Canada, Oct. 2000.

[4] C. Cachin, "An information-theoretic model for steganography" in Information Hiding, 2nd Int. Workshop (D.Aucsmith, ed.) vol. 1525 of Lecture Notes in Computer Sciences, pp. 306-318, 1988.

[5] B. Chen and G. W. Wornnell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding", IEEE Trans. On Info. Theory, vol. 47, no. 4, pp. 1423-1443, May 2001.

[6] K. Solanki, N. Jacobsen, S. Chandrasekaran, U. Madhow and B. S. Manjunath, "High-volume data hiding in images: Introducing perceptual criteria into quantization based embedding", ICASSP, May 2002.

[7] N. Jacobsen, K. Solanki, S. Chandrasekaran, U. Madhow and B. S. Manjunath,"Image adaptive high volume data hiding based on scalar quantization", IEEE Military Comm. Conf. (MILCOM), Oct.2002.

[8] M. H. Costa, "Writing on the dirty paper", IEEE Trans. On Info. Theory, vol. 29, no. 3, pp. 439-441, May 1983.

[9] P. Moulin and J. A. O'Sullivan, "Information-theoretic analysis of Information hiding", IEEE Trans. On Info. Theory, vol. 49, no. 3, pp. 563-593, May 2003.

[10] Y. Yunus, S. Ab Rahman, J. Ibrahim, " Steganography: A Review of Information Security Research and Development in Muslim World", American Journal of Engineering Research (AJER), Volume-02, Issue-11, pp-122-128, 2013.

[11] S. Mahajan, A. Singh, "A Review of Methods and Approach for Secure Steganography", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, pp-67-70, 2012.

[12] R. Chandramouli, N. Memon, "Analysis of LSB Based Image Steganography Techniques", in proceeding of IEEE ICIP, 2001.

[13] F. Abdullatif , A. W. Shukur, "Blind Color Image Steganography in Spatial Domain", Ibn Al- Haitham J. For Pure & Appl. Sci. Vol.24 (1), 2011.

[14] S. Atawneh, A. Almomani1, and P. Sumari, "Steganography in Digital Images: Common Approaches and Tools," IEEE Technical Review, Vol 30, Issue 4, 2013.

[15] S. Y. Ameen and M. R. Al-Badrany, "Optimal Image Steganography Content Destruction Techniques", Proceedings of the 2013 International Conference on Systems, Control, Signal Processing and Informatics, 2013.

[16] N. Hamid, A. Yahya, A. Badlishah, D. Najim and L. Kanaan, "Steganography in image files: A survey", Australian Journal of Basic and Applied Sciences, 7(1): 35-55, 2013.

AUTHOR PROFILE

The Author is Dr. Eng. Khaled N. ElSayed. He was born in Cairo, Egypt 9 Oct. 1963. He has got his PhD of computers and systems fromFaculty of Engineering, Ain Shams University, Cairo, Egypt, 1996. He has worked as an associate professor of computer science, in Umm-AlQura Uni. in Makkah, Saudi Arabia since 2006. Artificial Intelligence is his major. His interest research is Distant Education, E-Learning, and Agent.

Dr. Khaled N. ElSayed translated the 4th edition of "Fundamentals of Database Systems", Ramez Elmasei and Shamkant B. Navathe, Addison Wesley, fourth edition, 2004, published by King Saud University, Riyadh, Saudi Arabia, 2009. He is also the author several books in programming in C & C++, Data structures in C& C++, Computer and Society, Database Design and Artificial Intelligence.

A Minimum Number of Features with Full-Accuracy Iris Recognition

Ibrahim E. Ziedan
Dept. of computers and systems
Faculty of Engineering
Zagazig University
Zagazig, Egypt

Mira Magdy Sobhi
Dept. of computers and systems
Faculty of Engineering
Zagazig University
Zagazig, Egypt

***Abstract*—A minimum number of features for 100% iris recognition accuracy is developed in this paper. Such number is based on dividing the unwrapped iris into vertical and horizontal segments for a single iris and only vertical segments for dual-iris recognition. In both cases a simple technique that regards the mean of a segment as a feature is adopted. Algorithms and flowcharts to find the minimum of Euclidean Distance (ED) between a test iris and a matching database (DB) one are discussed. A threshold is selected to discriminate between a genuine acceptance (recognition) and a false acceptance of an imposter. The minimum number of features is found to be 47 for single iris and 52 for dual iris recognition. Comparison with recently-published techniques shows the superiority of the proposed technique regarding accuracy and recognition speed. Results were obtained using the phoenix database (UPOL).**

***Keywords*—Iris recognition; Iris features; Speed of Iris recognition; Features reduction**

I. INTRODUCTION

The purpose of 'Iris Recognition', a biometrical-based technology for personal identification and verification, is to recognize a person from his/her iris prints. In fact, iris patterns are characterized by high level of stability, distinctiveness and noninvasive nature. Each individual has a unique iris (as shown in Fig.1); and the difference even exists between identical twins and between the left and right eye of the same subject [3].

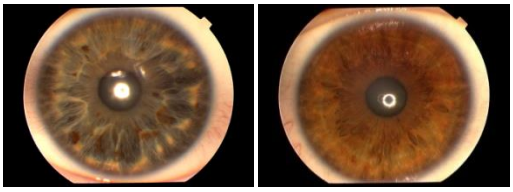


Fig. 1. Distinctiveness of human iris

Recently, iris recognition is becoming one of the most important biometrics used in recognition over fingerprints and facial recognition [2]. Facial recognition is relatively easy to fool. Age, facial hair, surgery, head coverings, and masks all may affect results. Fingerprints are not as accurate as iris recognition as they require physical contact with a scanner device that needs to be kept clean (hygiene issue) [7].

In view of the above, Iris recognition has attracted the attention of many researchers. Conventional techniques like Discrete Cosine Transform (DCT) and Haar Wavelet and a newly introduced one; column-means method were presented in detail in a previous publication by the authors [1].where accuracies of 98.44% and 97.66% were achieved.A preceding work by Aly I. Desoky et al. [4] used a technique based on template fusion of several iris images and achieved nearly 99% accuracy. However the paper is organized as follows:Section II is a general consideration one. Single Iris Recognition is considered in section III. Section IV discusses dual-Iris recognition.Performance results are introduced in section V. Section VI discusses the reduction of feature vector. Section VII presents a comparison between different techniques and paper conclusion is given in section VIII.

II. GENERAL CONSIDERATIONS

Features are extracted with different feature extraction methods to encode the unique pattern of the iris into a biometric template. The template that is generated in the feature encoding process may also need a corresponding matching metric, which gives a measure of similarity between two iris templates. This metric should give one range of values when comparing templates generated from the same subject eye, and another range of values when comparing templates created from different subject irises. These two cases should give distinct and separate values, so that a decision can be made with high confidence as to whether two templates are from the same subject iris, or from two different subjects.

The Euclidean Distance (ED) is employed for classification of iris templates [6]. Two templates are considered to be matching if the Euclidean Distance is lower than a specific threshold. As a result, a decision can be made in the matching step, based on threshold values. That is to say that a similarity between two iris images may be evaluated using the ED as compared to a threshold.

The proposed and employed methods may be described through two algorithmic steps. In the first step, an iris template for each image in the database is created and stored as shown in Fig.2. In the second step, an iris template for a query image is created and then a comparison based on ED is made. The system can accept or reject a subject according to the minimum value of ED, as shown in the flow chart of figure (3).

Step 1: Creating DB Templates

i. Transform the normalized area of the iris [5] into a rectangular block (unwrapped iris) of fixed dimensions and then normalize (normalization size=64×512)

ii. Compute Column-Means / Combined Rows & Column-Means for each image and store such Means coefficients in vectors with size n (512/576). This is the Features Vector (FV) of that image.

iii. Repeat items i and ii for every database image.

Step 2: Template of the Query Image

i. A feature vector for the query image may be formed in a similar way to that carried out in Step 1

ii. For a query image 'q' compute the Euclidean Distance (ED) to every database image 'p', using (1):

$$ED = \sqrt{\sum_{i=1}^{n}(Vp_i - Vq_i)^2} \quad (1)$$

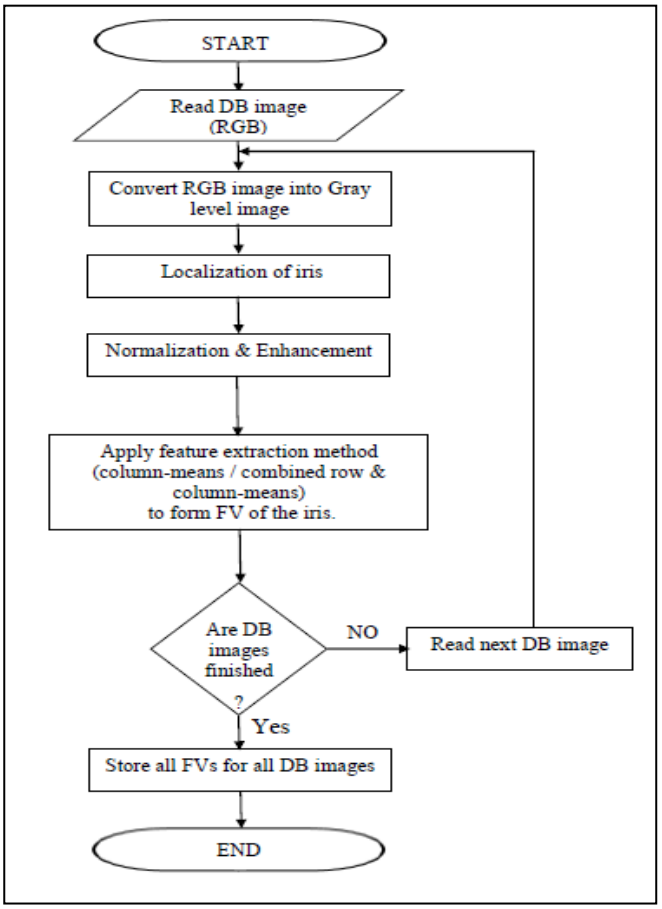


Fig. 2. Flow-chart of DB iris template creation

iii. Determine the DB image with minimum ED that is less than a threshold value. This corresponds to a matching

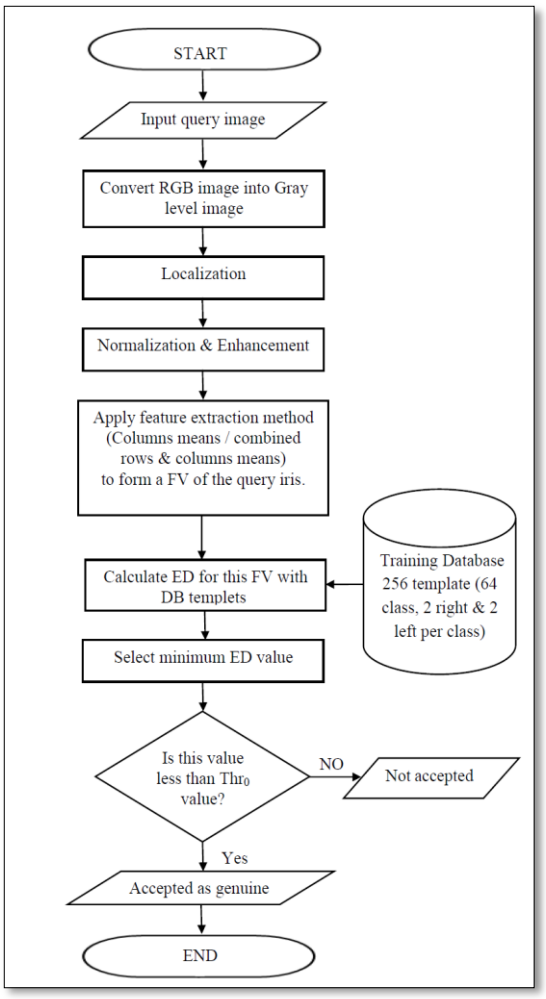


Fig. 3. Flow-chart of a subject identification

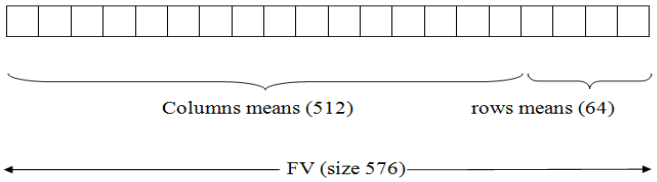


Fig. 4. FV of an eye template using combined rows & column-means method

between the two images p and q as shown in Fig. 5.

For the combined Rows & Column-Means method a FV of n=576 elements for each image may be formed as shown in Fig. (4).

During performance testing, a test image was considered and compared with all images in the database. The percentage of correct detections (genuine acceptances) is the percentage of relevant images returned and the percentage of incorrect detections is the amount of irrelevant images returned.

A threshold value must be chosen to determine the best accuracy and optimum (smallest), False Acceptances (FA) and False Rejections (FJ). As stated-above if the Euclidean Distance between two templates is less than the threshold value the templates would have been generated from the same iris and a match occurs (GA). On the other hand if the ED is greater than the threshold value the two templates are considered to have been generated from different irises.

It should be noted that smaller threshold values relate to higher rejections (i.e. less GA) of images belonging to DB subjects(i.e. FJ) while higher threshold values may cause higher false acceptances (FA) of impostor images

III. Single Iris Recognition

In a single iris recognition both the Column-Means method and the Rows and Column-Means method are adopted as given in section II. The definition of accuracy as employed in this study may be expressed as;

$$\text{Accuracy} = \frac{\text{GA}}{\text{GA+FA+FJ}} \qquad (2)$$

Where:

GA represents the no. of genuine acceptances,
FA represents the total no. of false acceptances and
FJ is the no. of false rejections.

This study is based on using phoenix DB where testing of 128 images with a no. of 256 DB ones was performed.

N.B. It should be noted that a query image of one subject iris may be regarded as an image of an imposter when excluding other images of the same subject from the DB set. Therefore the no. of false acceptances obtained may be considered as applied to both irises belonging to the DB and those of subjects from outside the DB (imposters) as well.

The identification accuracy obtained with combined rows & column-means method at different threshold values is shown in Fig. 5, where a maximum of 99.22% accuracy was achieved at a threshold value of 3.7.

Fig. 6 shows an example for the ED of a test image of index 1(right eye) to the 256 DB images with three threshold values (3, 5, and 3.7). One threshold value is small (3) and exhibits a false rejection of the test image. The second value is relatively high (5) and shows many false acceptances. This situation may be considered as resulting from imposter's irises as indicated above. The third value which is the optimum (3.7) shows a true matching between the test image and the two images of the same eye belonging to the same subject.

IV. Dual Iris Recognition

Iris recognition using both eyes of an individual has not been extensively investigated. However, for an iris recognition at a distance, capturing a good quality image of the same eye at different times is a challenging task and so a dual-iris approach is potentially beneficial.

For each subject, there are 4 images in the database; 2 left eye images and 2 right eye ones. Calling these images L1,

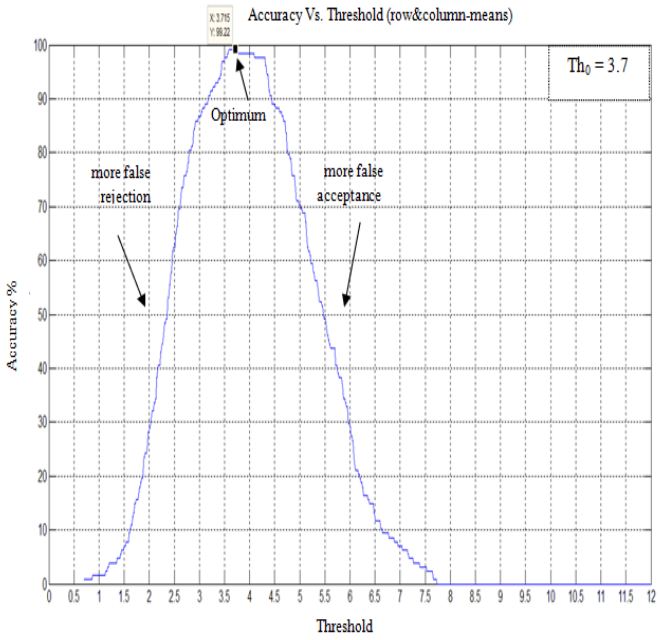


Fig. 5. Identification accuracy obtained for Combined Row & Column-Means Method with Threshold

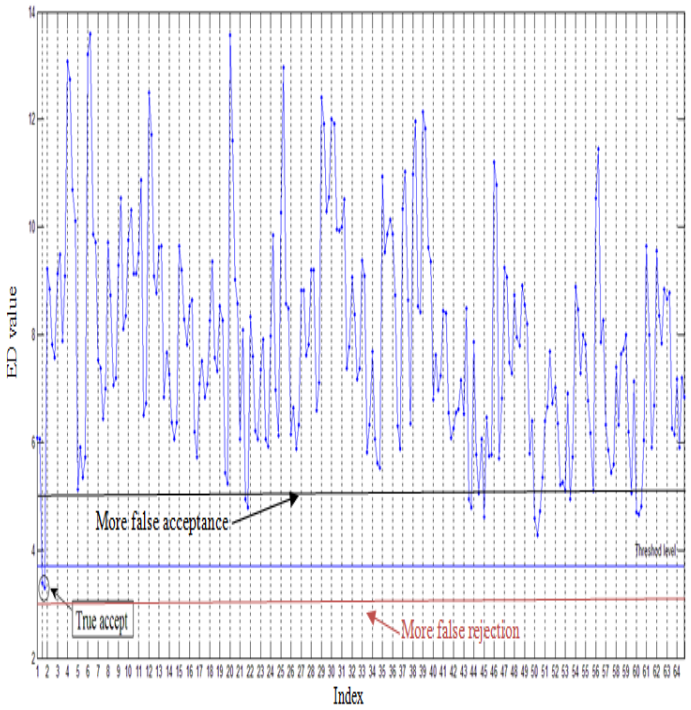


Fig. 6. Recognition of a test iris of index1with 256 DB images using Combined Rows & Column-Means Method

L2, R1 and R2, then four new templates may be formed for each subject denoted by L1R1, L1R2, L2R1 and L2R2. Hence

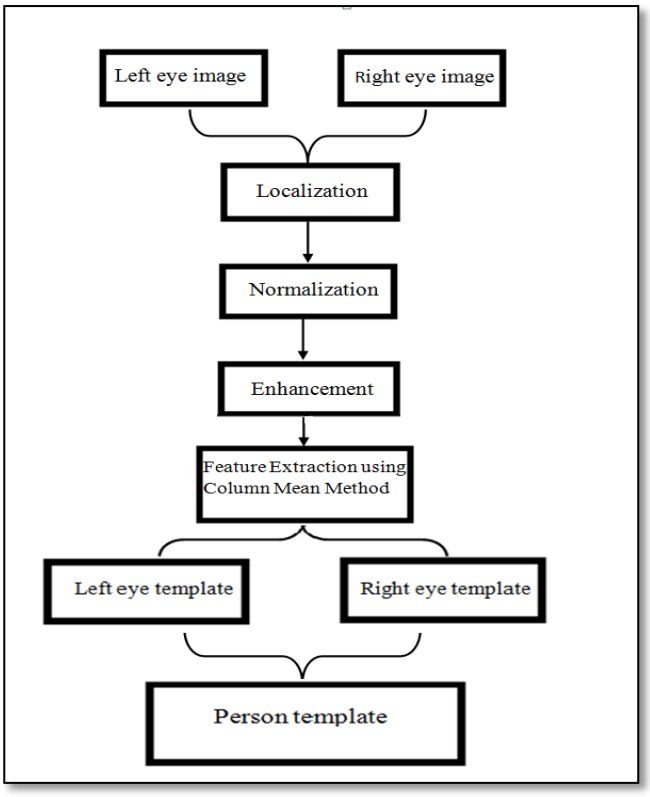


Fig. 7. Steps required for forming a subject's templates for dual iris recognition

test images contain one left eye and one right eye image that forms one template for each subject. Steps required for forming a subject's templates are shown in Fig. (7).

The mean of each column in the normalized image is computed for the two eyes. The resulting column-means form the FV of dual-iris images with a size of 1024 elements. Fig.8 shows the accuracy obtained with dual-iris recognition using column-means method at differentt threshold values.A 100% accuracy is shown at a threshold of 5.

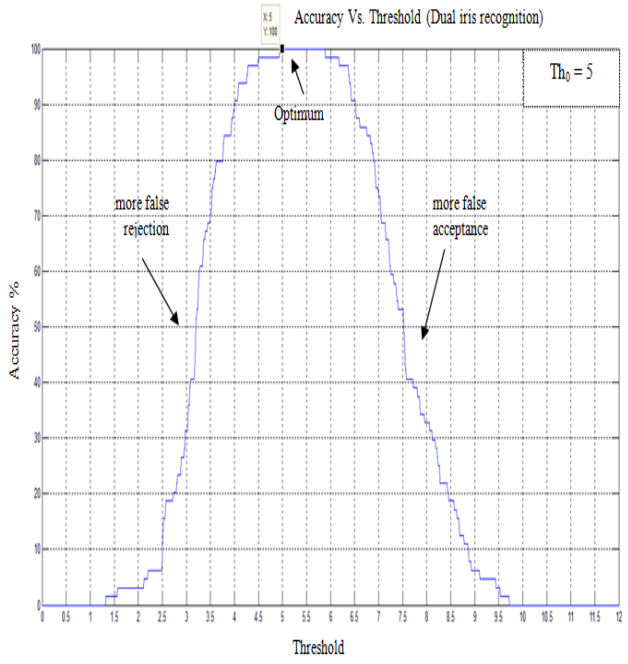


Fig. 8. Identification accuracy obtained for Dual -Iris recognition using Column-Means Method

V. Performance Results

A comparison between the recognition rate (accuracy) as obtained in the previously-published results [1] using Haar wavelet and DCT techniques and the proposed one are presented in Table I that shows also the corresponding optimum thresholds..

A comparison between Feature-Vector lengths for each method is presented in Table II.

As shown in Table I the feature-vector length of the Haar-Wavelet transform method is the smallest one, but the method gives the lowest accuracy.

VI. Reduction of Feature -Vector Length

A. Column-means method

To reduce storage requirements and improve execution speed the FV length ought to be reduced.

TABLE I. The Percentage of Recognition Rates

matching **Techniques**	**Recognition rate**	**Threshold**
Haar Wavelet Transform	97.66%	4.9
DCT(Discrete Cosine Transform)	98.44%	4.4
Column-Means Method	98.44%	3.7
Combined Rows & Column-Means Method	99.22%	3.7
Dual iris recognition using Column-Means method	**100%**	**5**

TABLE II. Features- Vector Lengths for Different Methods

Techniques	**F.V. length**
Haar wavelet transform	**480**
DCT	3072
Column-means method	512
Combined row& column-means method	576
Dual iris recognition using column-means method	1024

TABLE III. REDUCTION OF FEATURE-VECTOR LENGTH IN COLUMN-MEANS METHOD

No. of columns	FV length	No. of segments	Accuracy (Column-Means Method)
1	512	512[1 col.]	98.44%
2	256	256[2 col.]	98.44%
4	128	128[4 col.]	98.44%
8	64	64[8 col.]	98.44%
9	57	56[9 col.] +1 [8 col.]	98.44%
10	51	50[10 col.] +1 [12 col.]	98.44%
11	47	46[11 col.] +1 [6 col.]	98.44%
12	43	42[12 col.] +1 [8 col.]	98.44%
13	40	39[13 col.] +1 [5 col.]	97.66%
14	37	36[14 col.] +1 [8 col.]	96%
16	32	32[16 col.]	95.31%

The unwrapped iris is divided into vertical segments each comprising a number of columns. Tests were performed where the number of columns per segment is changed from 1 to 16. A feature may be specified in terms of the mean per segment instead of a column mean. An optimum (maximum) number of columns per segment may be reached while maintaining the highest accuracy of recognition which is 98.44% (Table I). Results are presented in Table III where the optimum number of columns per segment is 12. This means that the FV length is reduced to only 43 features.

B. Combined horizontal and vertical segments

The same method can be applied to decrease the feature vector length of a combined rows & column-means method. Dividing the unwrapped iris height (64 rows) into horizontal segments starting with one row per segment was done. Combining ED_1 of such horizontal segments with ED_2 of the vertical ones taking different weights w_1 and w_2 where $w_1+w_2=1$, the ED of two templates used for such method was calculated by using the formula:

$ED = w_1 \times ED_1$ (Row-segment means) + $w_2 \times ED_2$ (Column-segment means)

Weighting factors were changed so as to reach the best recognition rate with the minimum no. of segments vertically and horizontally (maximum no. of columns or rows per segment).

The results presented in Table IV show that the optimum values of weights w_1 and w_2 are 0.3 and 0.7 for the horizontal and vertical segments, respectively. This result is based on maintaining the accuracy at 100% while changing the number

TABLE IV. ACCURACY OF COMBINED HORIZONTAL & VERTICAL-SEGMENT MEANS METHOD AT DIFFERENT WEIGHTS

No. of Col.	No. of rows	Accuracy				
		$w_1=0.1$ & $w_2=0.9$	$w_1=0.2$ & $w_2=0.8$	$w_1=0.3$ & $w_2=0.7$	$w_1=0.4$ & $w_2=0.6$	$w_1=0.5$ & $w_2=0.5$
1	1	99.22	99.22	99.22	99.22	99.22
2	1	99.22	99.22	99.22	99.22	98.44
4	1	100	99.22	99.22	99.22	98.44
8	1	100	100	99.22	99.22	96.09
12	1	100	100	99.22	97.66	94.5
12	2	99.22	100	99.22	99.22	96.88
12	4	99.22	100	100	99.22	99.22
12	8	99.22	99.22	100	100	99.22
12	16	98.44	99.22	100	100	100

of columns and rows per segment. The minimum number of features was achieved with an optimum number of 12 columns per a vertical segment and 16 rows per a horizontal segment as shown in Table IV.

Accordingly the feature-vector length can be reduced using the principle of horizontal and vertical segments to 47 only instead of 576; as indicated in Table V.

C. Dual iris recognition based on vertical segments

The same method of vertical-segments division can be applied to dual-iris recognition based on column-means method. Originally the method has a feature vector length of.

TABLE V. FEATURE VECTOR LENGTHS IN COMBINED ROWS & COLUMN-SEGMENTS MEANS METHOD (W_1=0.3, W_2=0.7)

No. of columns/ segment	No. of rows/ segment	Feature vector length	Accuracy
1	1	576 (512+64)	99.22
2	1	320 (256+64)	99.22
4	1	192 (128+64)	99.22
8	1	110 (64+46)	99.22
12	1	107 (43+64)	99.22
12	2	75 (43+32)	99.22
12	4	59 (43+16)	100
12	8	52 (43+8)	100
12	16	47 (43+4)	100
12	32	45 (43+2)	97.66

1024 elements (512 for each eye). This feature vector length can be reduced using a segment-division technique as employed above. Tests were carried out while changing the number of columns per segment for both irises. An optimum (maximum) number of columns per segment were reached while keeping the accuracy of recognition at 100%. This number was found to be 20 columns per segment reducing the FV length to only 52 instead of 1024; as obvious in table (6).

VII. Comparison Between Different Techniques

Comparison between different techniques regarding accuracy, no. of features and speed of recognition is presented in table (7).

It is obvious from table (7) that 100% accuracy may be achieved with one iris using combined horizontal & vertical segments method in only 5.6 msec. for feature extraction and recognition of a test image. This is the best recognition method using one iris. These results are based on utilizing a machine with dual-core processor and a frequency of 2.7 GHz

TABLE VI. Feature-Vector Length in Dual Iris Recognition Based on Segment-Means Method

No. of columns	FV length	No. of segments	Accuracy
1	1024	2*(512[1 col.])	100%
2	512	2*(256[2 col.])	100%
4	256	2*(128[4 col.])	100%
8	128	2*(64[8 col.])	100%
16	64	2*(32[16 col.])	100%
17	62	2*(30[17 col.] +1 [2 col.])	100%
18	58	2*(28[18 col.] +1 [8 col.])	100%
19	54	2*(26[19 col.] +1 [18 col.])	100%
20	52	2*(25[20 col.] +1 [12 col.])	100%
21	50	2*(24[21 col.] +1 [8 col.])	98.44%

TABLE VII. Final Comparison Between the Proposed Methods and Others

Techniques	Recognition rate	FV length	Execution time of FV extraction& matching (msec.)
Wavelet	97.66%	480	20
DCT	98.44%	3072	135
Vertical segments Method	98.44%	43	5
Combined vertical & horizontal segments Method	100%	47	5.6
Dual iris Recognition based on vertical segments	100%	52	9.5

VIII. Conclusions

This paper introduced three different methods to enhance the performance of iris recognition system. The contribution aspects of this work included the enhancement of the iris recognition accuracy, the enhancement of the system's speed during feature-vector extraction stage and recognition stage. The enhancement is mainly a result of the reduction of feature-vector length. The first proposed method is the vertical segments-based features with accuracy of 98.44%. The second method is a combined vertical & horizontal segments-based feature with accuracy of 100%. A dual-iris recognition system based on vertical segments only gave a 100% recognition rate as well.

A smaller size of feature vector contributes to speeding up the matching stage. However, an optimal feature -vector length of 52 elements for dual-iris recognition and 47 elements for one iris with combined horizontal & vertical segment-means recognition was reached. Both achieved accuracy (recognition rate) of 100%.

Comparisons between the introduced approaches as regards accuracy, feature extraction and matching time and feature-vector length were presented in detail.

The recognition time of Dual-Iris method is approximately 75% more than that of the combined horizontal & vertical-segments method of one iris. Their values are 9.5 and 5.6 msec. respectively. Matching time only for both methods was found to be closer to each other (1.5and 1.1 msec.) than feature extraction time (8 and 4.5 msec.).

References

[1] Ibrahim Ziedan and Mira M. Sobhi, “Comparison between Haar wavelet transform,DCT and proposed column- mean method-based Iris encoders,” EIJEST, Vol.17, No.2, 2014.

[2] G. P. Khetri, V. P. Pawar, D. C. Jain "Human Computer Interface Through Biometric Iris Recognition System,"International Journal of Computer Science & Engineering Technology (IJCSET), Vol. 3, No. 7, July 2012.

[3] Karen Hollingsworth, Kevin W. Bowyer, Stephen Lagree, Samuel P. Fenker, Patrick J. Flynn, "Genetically Identical Irises Have Texture Similarity That Is Not Detected By Iris Biometrics," Computer Vision and Image Understanding, Volume 115, Issue 11, November 2011, Pages 1493–1502.

[4] Aly I. Desoky, Hesham A. Ali, Nahla B. Abdel-Hamid " Enhancing iris recognition system performance using template fusion" 10th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT) Luxor, Egypt, December 15–18, 2010

[5] L. Masek and P. Kovesi. MATLAB Source Code for a Biometric Identification System Based on Iris Patterns. The School of Computer Science and Software Engineering, The University of Western Australia. 2003.

[6] Ya-Ping Huang, Si-Wei Luo, En- Yi Chen, “An efficient iris recognition system”, International Conference on Machine Learning and Cybernetics, pp. 450-454, 2002.

[7] "Biometric Comparison Guide", **https://epic.org/privacy/surveillance/spotlight/1005/irid_guide.pdf, last access 6/1/2015.**

[8] Database: Iris database is available on http://phoenix.inf.upol.cz/iris/download/

Apply Metaheuristic ANGEL to Schedule Multiple Projects with Resource-Constrained and Total Tardy Cost

Shih-Chieh Chen*
Department of Information Management
Overseas Chinese University
Taichung, Taiwan, R.O.C.

Ching-Chiuan Lin
Department of Information Management
Overseas Chinese University
Taichung, Taiwan, R.O.C.

***Abstract*—In this paper the multiple projects resource-constrained project scheduling problem is considered. Several projects are to be scheduled simultaneously with sharing several kinds of limited resources in this problem. Each project contains non-preemptive and deterministic duration activities which compete limited resources under resources and precedence constraints. Moreover, there are the due date for each project and the tardy cost per day that cause the penalty when the project cannot be accomplished before its due date. The objective is to find the schedules of the considered projects to minimize the total tardy cost subject to the resource and precedence constraints. Since the resource-constrained project scheduling problem has been proven to be NP-Hard, we cannot find a deterministic algorithm to solve this problem efficiently and metaheuristics or evolutionary algorithms are developed for this problem instead. The problem considered in this paper is harder than the original problem because the due date and tardy cost of a project are considered in addition. The metaheuristic method called ANGEL was applied to this problem. ANGEL combines ant colony optimization (ACO), genetic algorithm (GA) and local search strategy. In ANGEL, ACO and GA run alternately and cooperatively. ANGEL performs very well in the multiple projects resource-constrained project scheduling problem as revealed by the experimental results.**

Keywords*—multiple project scheduling; resource-constrained project scheduling; ANGEL; ant colony optimization; genetic algorithms; local search; metaheuristics

I. INTRODUCTION

The resource-constrained project scheduling problem (RCPSP) is an important problem both in practice and research. Many researchers work on the single-project case for several years and have very good results, but the research works for the multiple-projects case are only a few. The multiple projects RCPSP model is a more realistic model.

The work by Lova et. al. [1] indicated that 84% of the companies, in the Valencian Region-Spain which responded to their survey, work with multiple projects. This data is in line with the work by Payne [6] that indicated that up to 90% of all projects occurred in the multiple-project context. And the due date and tardy cost are also important realistic situations to be considered. These reasons motivate us to research and to find some good algorithms on this topic.

We summarize some research works for the multiple projects RCPSP. Fendley [8] used multiple projects with 3 and 5 projects and concluded that the priority rule MINSLK is the best for the measurements project slippage, resource utilization and in-process inventory. Kurtulus and Davis [2] showed six new priority rules to the multiple projects instances they designed. They showed that the priority rules MAXTWK and SASP are the best when the objective is to minimize the mean project delay. Kurtulus and Narula [3] showed that the priority rule Maximum Penalty is the best to minimize the sum of the project weight delay. Dumond and Marbert [5] designed five resource allocation heuristics and four strategies to assign due dates to the projects. They showed that the priority rule FCFS with the Schedule Finish Time Due Date rule is the best to minimize the mean completion time, the mean lateness, the standard deviation of lateness and the total tardiness. Lova et al. [1] developed a multi-criteria heuristic to schedule multiple projects with the one-time criteria (mean project delay or multiple project duration) and one-no-time criteria (project splitting, in-process inventory, resource leveling or idle resources).

RCPSP has been proven to be an NP-Hard problem. Many evolutionary algorithms and metaheuristics were proposed to solve RCPSP and the related extension problems. Tseng and Chen [9] proposed an algorithm ANGEL which combines ant colony optimization (ACO), genetic algorithm (GA) and local search strategy (LS) to solve the single project RCPSP. Chen and Lin [13] proposed a discrete particle swam optimization to solve RCPSP. The experimental results of these works showed that they are compatible to other state-of-the-art algorithms in the literature for solving instance sets in PSPLIB [11]. Tseng and Chen [10] also proposed a two-phase genetic local search algorithm to solve the single project RCPSP with multiple modes. Chen [12] proposed a two-phase genetic local search algorithm to solve the single project RCPSP with generalized precedence constraints. Rivera et al. [7] proposed an algorithm which combined the GRASP, Scatter Search and justification to solve RCPSP. These researches showed that combined heuristics and evolutionary algorithms, like ANGEL, can solve these RCPSP and related extension problems efficiently.

The remaining parts of the paper are organized as follows. In Section II, we provide a description of the problem. In Section III, we describe ouer ANGEL algorithm. In Section IV,

the computational results are shown and in Section V the concluding remarks are given.

II. Problem Description

We consider a multiple project scheduling problem with n independent projects $P_1,\cdots, P_n$ where $d_1,\cdots, d_n$ and $c_1,\cdots, c_n$ are the due dates and tardy costs per day for the projects respectively. There are K kinds of common resources which are renewable that all projects share on them. For project P_i, $i=1, 2,\cdots, n$, the set J_i consists of num_i non-dummy activities and each activity has deterministic duration and resource demand for execution. By the single-project approach, we add two dummy activities as the source and the sink to bind the projects together. Hence this multiple-project problem can be considered as a single project problem in which the activity set J has $num=\sum_{i=1}^{n} num_i$ non-dummy activities. All non-dummy activities in J are renumbered from 1 to num sequentially. Activity 1 to num_1 are the activities of project P_1, activity num_1+1 to num_1+num_2 are the activities of project P_2 and so on. Activity 0 and activity $num+1$ are the source and the sink, the dummy activities. So $J=\left(\bigcup_{i=1}^{n} J_i\right)\cup\{\ 0,\ num+1\ \}$ is the set of activities. Fig. 1 shows a multiple projects instance with 2 independent projects and the corresponding combined single project is shown in Fig. 2, where d_j is the duration and r_{jk} is the resource demand for resource k of activity j.

Let $PSet_i$ be the set of all immediate predecessors (activities) of activity i. The precedence constraints are given that activity i cannot be executed before all activities belong to $PSet_i$ have finished. For resource k, the per-period-availability is given by a constant R_k. Each dummy activity has zero duration and does not require any resource.

In multiple projects RCPSP, let m_1, …, m_n be the makespans we scheduled for each project respectively, then the total tardy cost TC is defined in equation (1) as the sum of tardy costs of the projects which cannot be accomplished in their due dates. The objective of the problem considered in this paper is to minimize the total tardy cost (TC) subject to all the precedence and resource constraints.

$$TC=\sum_{m_i>d_i}(m_i-d_i)\cdot c_i \qquad (1)$$

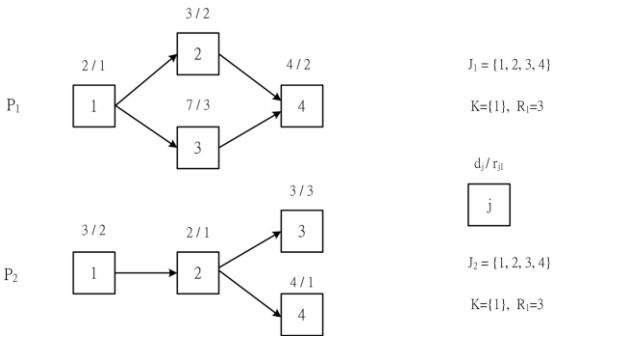


Fig. 1. A multiple projects instance

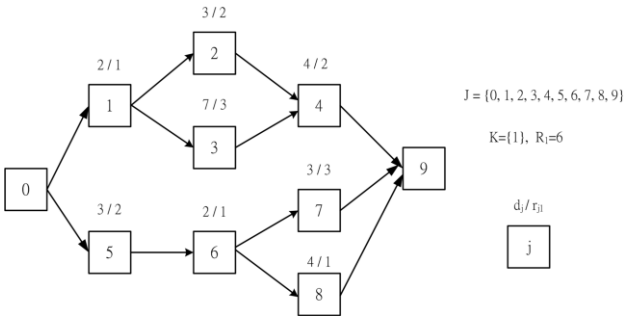


Fig. 2. The corresponding combined single project instance

In this paper, we use a precedence-feasible activity list to represent a solution. When a precedence-feasible activity list is given, we use the list scheduling method to create a schedule. To make a schedule, we apply either the forward scheduling or the backward scheduling on the activity list to set the execution starting time for each activity. The forward scheduling sets the execution starting time of an activity, from the front to the end of the list, as early as possible but satisfies the resource constraints. That is, if an activity lacks for some resources to start its execution, then its execution starting time will be delayed until some activity is finished and the resources are released to satisfy its resource demand. The backward scheduling sets the execution starting time of an activity, from the end to the front of the list, as late as its finish time is just right before the earliest execution starting time of all its successors. If an activity lacks for some resource for its execution, then the execution starting time will be set earlier just right before the starting time of some activity in order to satisfy its resource demand.

III. The ANGEL Metaheuristic for the Multiple Projects RCPSP

In this section, we present the strategies in ANGEL metaheuristic for the multiple projects RCPSP. We modify the algorithm of Tseng and Chen [12] because we solve this problem by single-project approach but coincide the

characteristic of this problem. ANGEL consists of the ACO, the GA and the local search strategy. All parts of the metaheuristic ANGEL are described in detail in the following.

A. The Ant Colony Optimization (ACO)

In original ACO several ants share the common memory set called *pheromone* and each ant find its own path of solution independently. The schemes *local updating*, *global updating*, and *evaporation* change the common memory by the experience of each ant, experience from global best solution and decreasing during time past respectively. We apply ACO to generate a population of precedence-feasible activity lists. To construct an activity list by a specific ant *x*, we first put the dummy activity 0 into the first position of the list. Then, if activity v is put in position j, ant x has to choose another activity from the candidate set N_j and put the chosen activity to position $j+1$ of the list. The candidate set N_j consists of the activities whose predecessors have been put in the list. When activity v in position *j*, the probability that ant x chooses activity w to be in position $j+1$ is defined in equation (2), where q_0 is an user-defined parameter, q is a random number drawn between 0 and 1, and τ_{vu} is the amount of pheromone been deposited on the ordered pair (v,u). S is a random variable selected according to the probability distribution given in equation (3), where $\tau_{min} = min_{u \in N_j}\{\tau_{vu}\}$.

$$w = \begin{cases} \arg \max_{u \in N_j}\{\tau_{vu}\} & , \text{ if } q \le q_0 \\ S & , \text{ otherwise} \end{cases} \quad (2)$$

$$p_{vw}^{x} = \begin{cases} (\tau_{vw} / \tau_{\min})^2 / \sum_{u \in N_j} (\tau_{vu} / \tau_{\min})^2 & \text{,if w} \in N_j \\ 0 & \text{,otherwise} \end{cases} \quad (3)$$

The formulae of the local updating, the evaporation, and the global updating are described in equations (4)-(6) respectively.

$$\tau_{vw} \leftarrow \tau_{vw} + \Delta\tau \quad (4)$$

$$\tau \leftarrow (1-\rho)\cdot\tau \quad (5)$$

$$\tau_{vw} \leftarrow (1-\alpha)\cdot\tau_{vw} + \alpha\cdot\Delta\tau_{vw} \quad (6)$$

In (4), $\Delta\tau$ is a small increment when the local updating is performed on the ordered pair (v,u). In (5), the evaporation means the amount of pheromone of all ordered pairs are decremented by a ratio ρ, where $0 < \rho < 1$. The increment $\Delta\tau_{vu}$ of the global updating in (6) is defined in (7), where $0 < \alpha < 1$ and TC_{gb} is the minimal total tardy cost of schedules ever found. Note that the global updating is only conducted on the list with the minimum total tardy cost in each ACO iteration.

$$\Delta\tau_{vw} = \begin{cases} 1/TC_{gb} & \text{if activity } w \text{ is next to activity } v \text{ in the list with the minimum total tardy cost found by ants in an iteration} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

By the effort of the ACO, the ants find for us a population of activity lists, then evaluate the associated schedules by forward and backward scheduling method, and evaluate the makespans and the corresponding total tardy costs. From the schedules conducted by forward and backward scheduling, only the one with smaller total tardy cost will be reserved. Tie will be broken by random selection.

B. The Genetic Algorithm (GA)

The GA we proposed is a permutation-based GA. The chromosomes in the population are the activity lists which consist of 90% constructed by the ACO and 10% randomly generated. The fitness of an activity list is the inverse value of the total tardy cost of the corresponding schedule. The crossover, mutation and selection operators are as follows.

We implement *two-point forward crossover* and *two-point backward crossover*, which are modified versions of the two-point forward-backward crossover proposed by Alcaraz and Maroto [4], in GA. The two-point forward crossover constructs the offspring from front to rear, and the two-point backward crossover constructs the offspring from rear to front. The crossover operators are defined as follows.

Two parent lists, called father and mother, produce two offspring, called son and daughter. We first randomly draw two crossover-points denoted by L_1 and L_2. To produce the son, when the two-point forward crossover is applied, the first positions of the son are directly taken from the first L_1 activities of the father. Then, in the father and the mother, the activities that have been taken are marked. The next $L_2 - L_1$ positions of the son are taken from the first $L_2 - L_1$ unmarked activities of the mother. In the father, these taken activities are marked. The rest positions of the son are taken from the rest unmarked activities of the father. All the activities taken from the mother or the father are in their relative order. The daughter is produced by interchanging the roles of the father and the mother. The two-point backward crossover works as a "reverse version" of the forward crossover that takes the activities and constructs the offspring from rear to front. To apply the crossover operators, the lists in the population are randomly divided into *pop*/2 pairs and a probability threshold, *pcro*, is specified. For each pair of lists, the two-point forward crossover is applied if the random number drawn is greater than the threshold. Otherwise, the two-point backward crossover is applied.

We design two mutation operators which try to pick out some activities and then randomly put them back as long as the precedence relations are satisfied. First, the activities in a list are classified to two classes A and B. Those activities in class A are picked out by a larger probability *pmut2* while those in class B by a smaller probability *pmut3*. When mapping a list to the corresponding schedule by the list scheduling method, an activity may not be started right after all its predecessors finished because lacking the resources it needs. We call this activity a *delayed activity*. If an activity in a list is a delayed activity while this activity is not a delayed activity in most of lists in the population, this activity belongs to class A in this list. Otherwise, activities not belonging to class A in a list

belong to class B. When applying the mutation to a list, for each activity in the list, if the activity belongs to class A, it is picked out by probability *pmut2*, otherwise, it is picked out by probability *pmut3*. In a random order, those pick-out activities are then randomly put back to the list as long as the precedence relations are preserved.

We implement the *ranking selection* and the *2-tournament selection* in our GA. After the crossover and the mutation, there are 2**pop* lists in the population, *pop* parent lists and *pop* offspring lists. In the ranking selection, we select the first *pop* lists from the population that is ranking by the makespan to construct the new population. In the 2-tourament selection, two lists are selected randomly from the population and the one with smaller makespan will be put in the new population. This procedure will be repeated *pop* times to construct the new population.

C. The Local Search Strategy

The local search strategy in this study is the forward-backward local search (FBLS) proposed by Tseng and Chen [9]. This local search utilizes the standard representation of permutation to reduce the search space and both forward scheduling and backward methods to improve the solution quality that very few computational effort is needed. The FBLS tries to search better solutions for a given permutation by following steps: (i) evaluate the forward schedule of the list, sort the operation starting times of activities and make the standard representation permutation the list by the order of operation starting times; (ii) evaluate the backward schedule of the list, sort the operation starting times of activities and then make a new permutation the list by the order of operation starting times; (iii) evaluate the forward schedule of the list, sort the operation starting times of activities and then make a new permutation the list by the order of operation starting times; compare the makespans of the schedules evaluated from the previous three steps and replace the list by the permutation which has the smallest makespan at last. From the experimental results conducted by Tseng and Chen [9], this local search is a very fast and effective local search to improve the solution quality in RCPSP.

D. The ANGEL

In the process of ANGEL, we apply the ACO first to generate activity lists, followed by applying the forward and the backward scheduling to each of them and reserve the better one. These lists along with several randomly generated lists are used as the initial population of GA. The local search is applied to the new lists to search better solutions. In GA, if the best schedule ever found is not improved for *GenStuck* generations, we apply the mutation operator and the 2-tournament selection to the population. And then start GA again. After applying the mutation operator *LoopStuck* times, it seems that the population is highly homogeneous, and then the ACO is applied again, construct new population, and begins another run of ANGEL. The procedure of ANGEL is shown in Fig. 3.

IV. Computational Results

We create eight sets of multiple projects instances, as shown in Table I, by combining single project instances from the PSPLIB. Each of the instance set J30 and J60 contains 480 single project instances and each instance contains 30 and 60 non-dummy activities. We combine the instances of J30 and J60 randomly to be the multiple projects problem instances.

```
Procedure ANGEL
  While ( !termination_angel )
    ACO, local search to the list, and generate the population
    While ( !termination_ga)
      Crossover
      Evaluate schedule, local search and local update pheromone
      Ranking selection
      If ( the best solution is improved ) global update pheromone
      If ( the best solution is not improved for a long time)
        Crossover
        Mutation
        Evaluate schedule and local search
        2-tournament selection
    End
    Global update pheromone
  End
```

Fig. 3. Procedure of ANGEL

TABLE I. Multiple Projects Instance Sets

Instance set	Instances	Projects in each instance
30_2	240	2 projects with 30 activities each
30_4	120	4 projects with 30 activities each
30_8	60	8 projects with 30 activities each
60_2	240	2 projects with 60 activities each
60_4	120	4 projects with 60 activities each
60_8	60	8 projects with 60 activities each
30_60_2_2	240	4 projects, 2 with 30 and 2 with 60 activities
30_60_4_4	120	8 projects, 4 with 30 and 4 with 60 activities

To show the effect of our method, we first define the upper bound for the instances. Suppose a multiple projects which consists of projects $P_1,\cdots,P_n$ where $d_1,\cdots,d_n$ and $c_1,\cdots,c_n$ are the due dates and tardy costs per day for the projects respectively. Let u_i be the best makespan when project P_i is scheduled as a single project RCPSP, then the upper bound for this instance is evaluated by equation (8).

$$UB = \sum_{i=1}^{n} (u_i - d_i) \cdot c_i \tag{8}$$

The statistical property of UB of each multiple project instance set is shown in Table II. As for the lower bound, it is obvious that zero is a trivial lower bound for each instance. We also define the improvement ratio IR in equation (9).

$$IR = \begin{cases} (\text{UB-TC})/\text{UB} \times 100\% & \text{if UB} \neq 0 \\ 100\% & \text{if UB} = 0 \ \& \ \text{TC} = 0 \\ -100\% & \text{if UB} = 0 \ \& \ \text{TC} \neq 0 \end{cases} \quad (9)$$

TABLE II. STATISTICAL PROPERTY OF UB IN EACH INSTANCE SET

Instance set	Max	Min	Ave.	S. Dev.
30_2	1334	0	175.59	263.38
30_4	1913	0	482.69	442.56
30_8	2047	52	801.32	485.14
60_2	3040	0	446.60	707.19
60_4	4076	0	906.26	1069.02
60_8	6204	0	1790.90	1478.86
30_60_2_2	3346	0	622.20	741.90
30_60_4_4	4308	4	1388.95	1102.76

For example, the 17th instance of the multiple projects instance set 30_2 consists of the 4th and 81st instances from the single project instance set J30. The best makespans of each project in single project RCPSP are 62 and 83. The due dates are 55 and 55, and the tardy costs are 28 and 19 per day for each single project respectively. Then

$$UB = (62-55)\times 28 + (83-55)\times 19 = 728$$

for the multiple project instance. The makespans for this multiple project instance when they are scheduled simultaneously are 55 and 62, then the total tardy cost

$$TC = (55-55)\times 28 + (62-55)\times 19 = 133$$

and the improvement ratio

$$IR = \frac{728-133}{728}\times 100\% = 81.73\%.$$

In our computational experiments, each instance set is tested 3 times and based on the average *IR*, the best case, the worst case and the average case are presented. Each instance is searched with 1000 or 5000 schedules evaluated. Table 4-10 show the computational results of *IR* and the percentage of the instances with zero total tardy cost ($TC = 0$) for all instance sets except the instance set 30_8. In instance set 30_8, the total tardy costs are zero for all instances within 1000 schedules.

TABLE III. COMPUTATION RESULTS OF INSTANCE SET 30_2

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	0	95.40	11.78	76.25
	Worst	100	0	95.36	11.96	76.25
	Average	100	0	95.38	11.90	76.67
5000	Best	100	0	96.61	10.29	80.42
	Worst	100	0	96.43	10.54	79.17
	Average	100	0	96.49	10.46	80.00

TABLE IV. COMPUTATION RESULTS OF INSTANCE SET 30_4

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	79.93	99.25	2.98	89.17
	Worst	100	79.17	99.20	2.94	90.00
	Average	100	79.13	99.23	2.99	89.44
5000	Best	100	86.13	99.68	1.70	92.50
	Worst	100	85.36	99.64	1.75	90.83
	Average	100	86.68	99.67	1.65	91.67

TABLE V. COMPUTATION RESULTS OF INSTANCE SET 60_2

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	-100	88.80	28.25	74.58
	Worst	100	-100	88.04	30.98	73.33
	Average	100	-100	88.29	30.07	73.89
5000	Best	100	2.84	93.71	15.52	78.33
	Worst	100	0	93.43	16.43	78.33
	Average	100	0.95	93.52	16.12	78.33

TABLE VI. COMPUTATION RESULTS OF INSTANCE SET 60_4

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	65.68	98.07	6.05	83.33
	Worst	100	57.73	97.73	7.51	85.00
	Average	100	59.92	97.91	6.70	84.44
5000	Best	100	81.04	99.13	2.95	86.67
	Worst	100	76.32	98.97	3.83	87.50
	Average	100	78.57	99.03	3.46	87.23

TABLE VII. COMPUTATION RESULTS OF INSTANCE SET 60_8

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	86.17	99.51	1.96	88.33
	Worst	100	80.81	99.42	2.59	88.33
	Average	100	84.27	99.46	2.24	88.33
5000	Best	100	98.66	99.97	0.19	95.00
	Worst	100	97.51	99.95	0.33	95.00
	Average	100	98.10	99.96	0.25	95.55

TABLE VIII. COMPUTATION RESULTS OF INSTANCE SET 30_60_2_2

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	59.99	98.93	4.32	87.92
	Worst	100	66.43	98.82	4.46	87.08
	Average	100	63.72	98.87	4.36	87.36
5000	Best	100	82.75	99.61	1.98	93.33
	Worst	100	79.13	99.53	2.19	92.92
	Average	100	80.79	99.58	2.09	93.05

TABLE IX. COMPUTATION RESULTS OF INSTANCE SET 30_60_4_4

Schedules	Case	*IR* (%)				$TC = 0$ (%)
		Max	Min	Ave.	S. Dev.	
1000	Best	100	94.40	99.82	0.80	93.33
	Worst	100	96.31	99.82	0.68	91.67
	Average	100	95.46	99.82	0.75	92.50
5000	Best	100	99.18	99.99	0.11	98.33
	Worst	100	99.59	99.99	0.05	97.50
	Average	100	99.29	99.99	0.09	98.06

We can see from all the tables that the average *IR* ratio and the percentage of instances with zero total tardy cost increase, and the standard deviation of *IR* ratio decreases as the number of schedules evaluated increases. These results means the total tardy cost of multiple projects will be improved effectively by ANGEL if more searching is conducted. We can also observe that if more projects are to be scheduled simultaneously, there are greater chances that projects be accomplished in their due dates. This result also fits the realistic situation and suggests that in a company, all projects that share the common resources should be scheduled simultaneously.

V. CONCLUDING REMARKS

ANGEL had been applied to solve the single project RCPSP [9] and the single project RCPSP with multiple modes [10] and obtained good results. In this paper we consider the problem that multiple projects sharing common resources are to be scheduled simultaneously subject to the precedence and resource constraints. The objective is to minimize the total tardy cost of the projects. The computational results show that ANGEL is effective on this problem. It also reveals that projects sharing common resources should be scheduled simultaneously rather than scheduled one by one.

From the computational results and other researchers' works we can find that the combined algorithms or metaheuristics performed well in solving discrete combinatorial optimization problems. The further researches of us are to test the combination of different evolutionary algorithms, like GRASP or Particle Swarm Optimization, to find better algorithms to different optimization problems

REFERENCES

[1] A. Lova, C. Maroto and P. Tormos, "A multicritertia heuristic method to improve resource allocation in multiproject scheduling," Euro. J. Oper. Res., vol. 127, pp.408-427, 2000

[2] I. S. Kurtulus and E. W. Daivs, "Multi-project scheduling: categorization of heuristic rules performance," Mana. Sci., vol. 28, pp161-172, 1982.

[3] I. S. Kurtulus and S.C. Narula, "Multi-project scheduling: analysis of project performance,: IIE Trans., vol. 17, pp.58-66, 1985.

[4] J. Alcaraz and C. Maroto, "A robust genetic algorithm for the resource allocation in project scheduling," Ann. Oper. Res. Vol. 102, pp.83-109, 2001.

[5] J. Dumond and V. A. Marbert, "Evaluating project scheduling and due date assignment procedures: An experimental analysis," Mana. Sci., vol. 34, pp.101-118, 1998.

[6] J. H. Payne, "Management of multiple simultaneous projects: A state-of-the-art review," Inter. J. Proj. Mana., vol. 13, pp-163-168, 1995.

[7] J. Rivera, L. Velásquez, F. Serna and G. Zapata, "A Hybrid Heuristic Algorithm for Solving the Resource Constrained Project Scheduling Problem," Revi. EIA, vol. 10, pp.87-100, 2013.

[8] L. G. Fendley, "Towards the development of a complet multiproject scheduling system," J. of Indu. Engi. Pp.505-515, 1968.

[9] L. Y. Tseng and S. C. Chen, "A hybrid metaheuristic for the resource-constrained project scheduling problem," Euro. J. Oper. Res., vol. 175, pp.707-721, 2006.

[10] L. Y. Tseng and S. C. Chen, "Two-Phase Genetic Local Search Algorithm for the Multimode Resource-Constrained Project SchedulingG. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[11] R. Kolisch and A. Sprecher, "PSPLIB – a project scheduling problem library," Euro. J. Oper. Res., vol.96, pp205-216, 1996.

[12] S. C. Chen, "A Genetic Local Search Algorithm for the Resource-Constrained Project Scheduling Problem with Generalized Precedence Constraints," The 6th IEEE Inter. Conf. Ubi-Media Comp., Aizu-Wakamatsu, Japan, November, 2013.

[13] S. C. Chen, "A Discrete Particle Swam Optimization for Scheduling Projects with Resource Constrained," The 2nd Inter. Conf. Comp., Meas., Cont. and Sens. Net., Tunghai University, Taiwan, May, 2014.

Development and Role of Electronic Library in Information Technology Teaching in Bulgarian Schools*

Tsvetanka Georgieva-Trifonova
Department of Mathematics and Informatics
University of Veliko Tarnovo
Veliko Tarnovo, Bulgaria

Gabriela Chotova
Professional School of Electrical and Electronics
Gorna Oryahovitsa
Bulgaria

***Abstract*—The electronic library can be considered as an interactive information space. Its creation substantially supports the communication between the teachers and the student, as well as between the teachers and the parents. The enlargement of information space allows improving the efficiency and the quality of teaching, assigning more projects for realization.**

The main purpose of this paper is to examine the role of the electronic library in teaching of information technologies in Bulgarian schools for providing more time for applying the learned material in order to increase the effectiveness of the educational process. We summarize and represent the advantages and disadvantages of the use of digital libraries in teaching information technologies together with the main features of a developed electronic library for teaching and educational subsidiary materials.

***Keywords*—electronic library; information technology teaching; multimedia information system**

I. Introduction

In recent years, there is in particular a lot of talk about how interactive teaching methods of the education help to students' motivation. Today students, and not only them, are in direct access to digital technologies in every aspect of their lives. It is clear that this has a huge impact on the personality of the student, his or her behavior and way of thinking. As to the educational system, developing of the new technologies passes with a speed higher than "escaping of the students from the lessons" and provide many more opportunities at the same time – many more challenges to the abilities of the teachers.

The modern innovative pedagogical technologies implement new training model based on complex information interactions between teachers, students and the means of information and communication technologies. Electronic technologies are implicated throughout the learning process – during the preparation and presentation of some information. The teacher remains to be a leading figure in structuring and preparing of the educational content. The main purpose of this paper is to examine the role of the electronic library in teaching information technologies in order to provide more time for the implementation of the material being learnt, and to increase the effectiveness of the educational process.

*The research is financed by project №09-590-13/10.04.2013, Integrated electronic services for the citizens and the business of St.Ciryl and St.Methodius University of Veliko Turnovo.

In the following sections, the advantages and disadvantages of the use of digital libraries in teaching information technologies are summarized and there are presented the main features of developed electronic library for teaching and educational subsidiary materials.

II. The Role of Electronic Library in Teaching Information Technologies

This study is motivated by observations and experience of one of the authors as a teacher of informatics, information technologies and vocational subjects in secondary school. These items are under specialty "System Programming", profession "Programmer". The authors of this paper notice some certain characteristics, related to school work and summarize them as follows:

- Too many hours are set aside to receive theoretical knowledge than for their consolidation with the help of their practical application;
- Lessons for acquiring practical skills are much less compared to those for obtaining of theoretical knowledge;
- As a result, the teacher is not always able to ensure consideration of all problems of a topic or a task.

These conclusions are the main motivation for the development of a web-based multimedia information system, which should serve as an electronic library for teaching and educational subsidiary materials. The main purpose of the proposed system is to allow input, storage, search and retrieval of the materials related to the themes of the specific subjects. In this way the students more easily can find the relevant information of a topic for a particular class. They have access to the materials in class, allowing more time for discussion on the topic. In parallel with the observation of the topic, there can be discussed concrete examples and to solve various tasks. As a result, more time for acquisition of the practical skills and skills for teamwork is provided.

The main purpose of the use of virtual environment for representing the teaching materials is to support the traditional forms of training. Its achievement sets the following requirements, which the development of an electronic library of teaching and educational subsidiary materials has to meet:

- Developing a multimedia information system that serves as an electronic library of educational and educational subsidiary materials;

The electronic library can be seen as a kind of multimedia system for retrieving information (multimedia information retrieval system) [1].

- The users (students) should have free access to the materials, without registration;
- Providing an access to teaching and educational subsidiary materials such as:
  - Homework – assignments and their solutions;
  - Example of variants of tests;
  - Projects – developments of students;
  - Thesis of students;
  - Lessons containing information that complements the mandatory educational content;
  - Other materials.
- Providing an opportunity for input and storage to:
  - Title, authors, type, description (abstract), keywords;
  - Documents (.doc, .pdf), presentations (.ppt, .pdf), images, audio, video, databases, programs of the studied programming languages, etc.
- Being able to perform searching for materials by:
  - Title, authors, type, description (abstract), keywords.

III. Advantages and Disadvantages of the Use of Electronic Library in Teaching Information Technologies

The main advantages of using an electronic library in teaching information technologies, which can be mentioned, are:

- The users (students) have access to pre-prepared training material at a time, convenient for them;

The system allows students to use it at a time, convenient for them. Moreover, the students can view and absorb the information in a manner and at a speed of their own pace and to communicate with the teacher.

- There is a free eligibility of training modules;
- There is given a fast and easy access to materials containing theoretical presentation of a topic, as well as the materials, presented practical implementation of the issues, being observed;
- The teacher has more time for practical training;

Materials stored into the multimedia information system, help or replace part of theoretical training;

- Multimedia information system helps to organize and conduct training sessions more dynamic;
- There is additional motivation of students for better and more completed presentation of their workings out;
- It helps to increase the interest and involvement of the students;
- It involves the skills of students to work in a team;
- There is an emphasis on the activities of the students;
- The teacher is a mentor;
- Training materials, available through the system, cover all subjects being taught.
- We have also noticed the following disadvantages:
- It requires an access to a computer and Internet;
- A secure connection is necessary;
- The connection is no always fast enough;
- It requires time and effort to develop and maintain the electronic library.

IV. Review of Existing Electronic Libraries for Teaching and Educational Subsidiary Materials

We conduct a research of the existing web-based systems that provide access to teaching and educational subsidiary materials and are supported by teachers in Bulgarian schools. Their options are briefly presented below:

- Electronic teaching materials of the Technological School "Electronic Systems" at the Technical University, Sofia city [2];

This website is managed by teachers and offers access to lectures and other materials of software specialties in Technological School "Electronic Systems". The proposed materials are designed to help to train students in subjects: Modular Java (Java and OSGI); Object-oriented programming (C ++); Python/Introduction to Programming; Software; Robotics; Database management system; Programming technology; Operating systems; Internet programming, etc. There are given assignments for thesis and requirements to them. There is not access available to developments of the students – assignments or thesis. This site is managed by Wordpress.

- Electronic library of the High School "Raycho Karolev", Gabrovo town [3];

The electronic library contains electronic tutorials, presentations and training projects concerning subjects of natural sciences and humanities cycle, as well as those of the initial stage of the primary education. The website implementation is based on PHP, there are created separated pages for each class, which lists topics for different subjects and materials for them. Besides lessons, there are offered tasks for exercises, assignments for projects, but there is a lack of their solutions and developments. It is impossible to carry out any searching.

- Electronic teaching materials of Musa Musa, a teacher of informatics and information technologies at the High School "Priest Paisii", Kardzhali town [4];

The website is implemented by using PHP and represents the distribution of lessons (presentations (.ppt) and additional materials (.rar)) on the subjects: Information Technologies; Informatics; Computer networks; Photoshop; Corel Draw; Web design; C #; C ++ and others. The access to them is free and also contains a test system. The website does not provide the ability to search and does not offer projects developed by the students.

The author of the website has used the following software tools: HTML, CSS and PHP for the test system, as well as animated gif files.

- Electronic teaching materials of the High School "Nayden Gerov", Lom city [5];

The website contains lessons and tasks for C ++ and Pascal. It is created for publishing tasks and their solutions. In order to publish the solutions of the tasks, the students must register themselves. Solutions are visible for everybody after an approval. The website provides the ability to search, but the result is only concerning the tasks for exercise. Lessons are not included into the search. This website is managed by Wordpress.

- Electronic teaching materials of the High School "Dimitar Blagoev", Provadia town [6];

The website is created to help to the students in their training in informatics and information technologies. It contains a variety of materials, divided into classes. Much of the materials except in HTML format are published in pdf format. Only after registration the user has access to the sections "Informatics - Other", "IT - Others", "Other subjects" and "Projects of the students", as well as the ability to write comments to some articles. This website allows users to search by keywords in HTML documents.

- Electronic teaching materials on informatics and computer architectures by Dian Pechenyashki – teacher of informatics in the High School of Economics, Veliko Tarnovo town [7];

The website is designed to assist the preparation and self-training of all students, studying informatics and computer architectures, in upper secondary education. For some lessons in the "Tests" chapter, there have been developed tests on subjects. The website does not provide the ability to search and does not offer students' projects. It is implemented by means of HTML, CSS.

- Electronic teaching materials, related to informatics and information technologies by Diana Martynova – a teacher at the Language School "Prof. Dr. Assen Zlatarov", Veliko Tarnovo town [8];

This website contains materials on informatics, information technologies, computer graphics and computer animation. For each subject, except the particular lesson, there are also exercises and tests.

The website does not provide the ability to search and does not offer projects developed by the students. It is implemented by using the platform alle.bg.

- Electronic teaching materials on informatics and information technologies by Mariana Bankova – a teacher at the Professional High School of Electrical Engineering, Varna town [9];

The website contains materials on informatics, programming, programming part I, programming part II, and information technologies. To each section of the subject there is homework. The website does not provide the ability to search, does not offer development of students. This website is made using a Website Builder, available from http://www.ucoz.com.

- Electronic teaching materials on informatics and information technologies by Zhivka Zhekova – a teacher at the High School with teaching foreign languages "A. S. Pushkin", Varna town [10].

The website contains materials on informatics, information technologies and web design. The website does not provide the ability to search and does not offer students' projects. It is implemented as a static HTML page that provides access to presentations of lessons.

After researching of the current state of the implementation of web-based multimedia information systems for e-learning materials and their use by students, it can be concluded that there is an increasing of their applying in the learning process. The analysis shows the possibilities for development and improvement of the existing systems, which would help to maximize the benefits of their applying.

V. Development of an Electronic Library for Teaching and Educational Subsidiary Materials

An electronic library for teaching and educational subsidiary materials in information technologies for secondary school is developed. It contains all the content, taught by one of the author' disciplines. It is available at: http://e-classroom.site88.net.

In Figure 1, we present UML diagram for the specific library. The main participants are the teacher and the students. The interface proposes the following functions:

- Free browsing and viewing the materials by subjects;
- Unlimited accessing to the materials;
- Searching for materials;
- Downloading the materials;
- Log into the system only allowed for the teacher;
- Adding data material is carried out only by the teacher;
- Adding a file only by the teacher;
- Feedback.

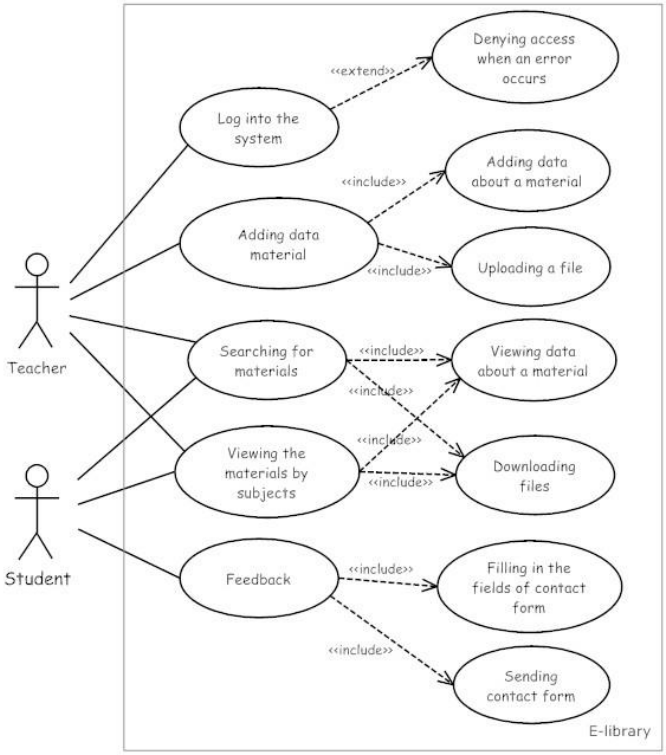


Fig. 1. Interface with UML diagram

We have designed and implemented a relational database [11, 12] in order to store the necessary data (materials, authors, subjects, keywords, categories of material, files for viewing and downloading).

A. Data modeling

In this section, the methodology of the designing the database `ELibraryDB`, storing the information about the materials, is represented. The main stages of the process of designing the database are [11, 13]:

- conceptual designing the database – creation of the conceptual data model which is completely independent from the details related to the implementation;
- logical designing the database – transformation of the conceptual model into the logical model;
- physical designing the database – implementation of the database with the means of a chosen database management system.

1) Conceptual designing

For the conceptual design of the database `ELibraryDB` we utilize the entity-relationship model, introduced in [14]. The following entity sets are established:

- Materials;

This entity set possesses two attributes: the identifier – *MaterialID*, the material's title – *Title*.

- KeyWords;

With the entity set *KeyWords* the following attributes are assigned: the identifier – *KeyWordID*, the keyword – *KeyWord*.

- Authors;

The attributes of this entity set are: the identifier – *AuthorID*, the author's name – *AuthorName*, the author's post – *Title*.

- Subjects;

With the entity set *Subjects* the following attributes are assigned: the identifier – *SubjectID*, the subject's name – *SubjectName*.

- *MaterialCategories* – the category of the material;

The attributes of this entity set are: the identifier – *MaterialCategoryID*, the category's name – *MaterialCategory*.

- *MaterialFiles* – the files for materials;

This entity set possesses two attributes: the identifier – *FileID*, the file location – *FilePath*.

- *FileCategories* – the categories of files for materials;

With this entity set the following attributes are assigned: the identifier – *FileCategoryID*, the file category's name (e.g. document, presentation, image, audio, video) – *FileCategory*.

- Users.

The attributes of this entity set are: the identifier – *UserID*, the user's name –*Username*, the user's password – *Password*.

The following relationships are defined:

- *A_M* – joins entities from the set *Authors* with entities from *Materials* and represents the authors of the materials;
- *M_K* – joins entities from the set *KeyWords* with entities from *Materials* and describes the keywords of the materials;
- *M_S* – joins entities from the sets *Materials* and *Subjects* and represents the subjects of the materials;
- *M_MS* – joins entities from the sets *Materials* and *MaterialCategories* and represents the category of the materials;
- *M_M* – joins entities from the sets *Materials* and *MaterialFiles* and represents the location of the materials;
- *F_MF* – joins entities from the sets *FileCategories* and *MaterialFiles* and shows the categories of the materials' files;
- *A_U* – joins entities from the sets *Authors* and *Users* and describes the user names of the materials' authors.

The entity-relationship diagram of the database `ELibraryDB` is shown in Figure 2. The entity sets are depicted as the rectangles, their attributes – as ellipses, the relationships – as rhombs.

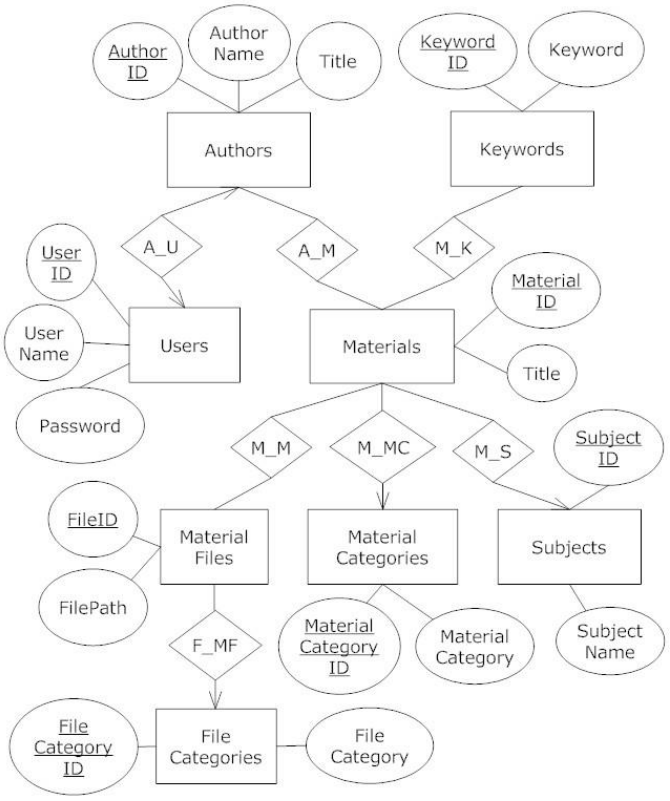


Fig. 2. ER diagram of the database ElibraryDB

2) Logical designing

The relation tables obtained after the transformation of the entity-relationships diagram into the relational model are presented in Figure 3.

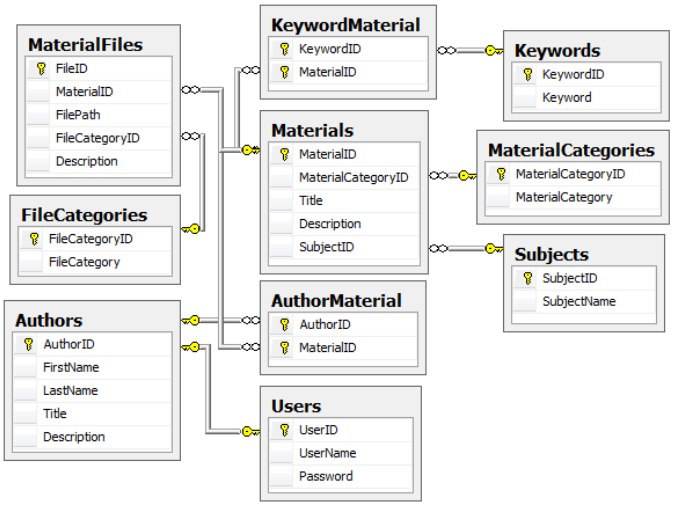


Fig. 3. Relational model of the database ElibraryDB

The proposed relational model allows a future development of the database – creation of new tables and/or attributes for storing additional information.

3) Physical designing

The database is implemented by means of the database management system MySQL. The structure of the database is defined to provide the best efficiency of the most frequently used operations – insertion, updating, data searching.

B. Description of the electronic library

The interface for the end user of the multimedia information system is implemented by means of HTML, CSS, PHP and offers the following features:

- Searching into all text attributes – title, author's name, type, description (summary) keywords;

When specifying a particular word (a symbol string) by the user, the found data are visualized and displayed, by using hyperlink to all materials that meet the specified word. Passed string is searched as a substring in the specified text attributes, describing the material.

- A feedback form is implemented;
- Data entry by the system administrator.

For this purpose, a login form is created. When a user enters the username and the password correctly, a new page for input the data for a material opens.

The multimedia information system is under development and has a certain level of completion. The future work includes:

- Advanced searching to set different criteria for different attributes, characterizing materials;
- Opportunity for online viewing (or listening) video (or audio) files tutorials, presentations of works by students, etc.

The proposed system and all materials accessible through it, are intended primarily for students in Bulgarian secondary schools. Therefore, the interface and the materials themselves are in Bulgarian.

VI. CONCLUSION

The proposed electronic library aims to visualize the teamwork between the students and the teacher leadership skills – to be able to organize and motivate students. Nowadays, the teacher should to stimulate and motivate [15], he or she should be a mentor, he or she should be able to organize different games to exercises students, to get the best results out of them. The teacher has to teach them to present their work and to develop their social skills. At school strength is that the students are all together, they can work in a team, the teacher is there to give them aguidance.

The data storage for the materials in the database provides opportunities for future work related to the collecting of information concerning the count and ratings of downloaded materials and their analysis by subjects, authors, keywords, type, etc. with means of tools for data processing, analysis and mining [16]. Similar analysis could assist to improve the quality of the materials in a way that they have to be useful, as well as to arouse an interest.

REFERENCES

[1] J. C. Nordbotten, "Multimedia information retrieval systems", 2008, available at: http://nordbotten.com/ADM/ADM_book/MIRS-frame.htm (accessed 17 February 2015).

[2] Technical school of Electronic Systems to Technical University, Sofia, „Software training in Technical school of Electronic Systems to Technical University: Lectures and other materials on software specialties", 2004-2015, available at: http://lubo.elsys-bg.org (accessed 17 February 2015).

[3] High School "Raycho Karolev", Gabrovo, "Electronic library", 2013, available at: http://rk-biblioteka.zymichost.com (accessed 17 February 2015).

[4] M. Musa, "Electronic teaching materials: Lessons and exercises on informatics and information technologies", High School "Father Paisii", Kardzhali, 2007, available at: http://www.paisii-kardjali.com/uroci/index.php (accessed 17 February 2015).

[5] High School "Nayden Gerov", Lom, "Electronic teaching materials: Tasks in Nayden Gerov", 2010-2015, available at: http://zadachi.gymnasium-lom.com (accessed 17 February 2015).

[6] High School "Dimitar Blagoev", Provadia, "Electronic teaching materials: Resources on informatics and information technologies", 2007, available at: http://www.it.souprovadia.info/?q=node (accessed 17 February 2015).

[7] D. Pechenyashki, "Electronic teaching materials on informatics and computer architectures", High School of Economics, Veliko Tarnovo, 2006, available at: http://www.iit.net-bg.info/inf.html (accessed 17 February 2015).

[8] D. Martynova, "Electronic teaching materials related to informatics and information technologies", Language School "Prof. Dr. Assen Zlatarov", Veliko Tarnovo, 2014, available at: http://egvt.alle.bg (accessed 17 February 2015).

[9] M. Bankova, "Electronic teaching materials on informatics and information technologies", Professional High School of Electrical Engineering, Varna, 2015, available at: http://bankova.ucoz.com (accessed 17 February 2015).

[10] Z. Zhekova, "Electronic teaching materials on informatics and information technologies", High School with teaching foreign languages "A. S. Pushkin", Varna, 2005, available at: http://jemil.my.contact.bg/it.htm (accessed 17 February 2015).

[11] H. Garcia-Molina, J. D. Ullman, and J. Widom, Database Systems: The Complete Book, Williams, 2002.

[12] A. M. Langer, Analysis and Design of Information Systems, Springer, 2008.

[13] T. M. Connolly and C. E. Begg, Database systems: a practical approach to design, implementation, and management, Addison-Wesley, 2005.

[14] P. Chen,"The Entity-Relationship Model: Toward a Unified View of Data", ACM Transactions on Database Systems, Vol.1, No.1, 1976, pp. 9-36.

[15] N. Nenkov and I. Ibryam, "Incentives and motives for studying Hadoop, HBase, MongoDB, Cassandra, NoSQL in subject Databases and Applications of college specialty Informatics and Information Technologies", In Proceedings of Mathematics, Computer Science and Computer information technologies, vol. 1, University Publishing House "K. Preslavski", 2012, pp. 398- 401 (in Bulgarian).

[16] N. Nenkov and I. Ibryam, "A survey of the open source platforms Rapidminer and Konstanz Information Miner (KNIME) for data processing, analysis and mining", In Proceedings of Pedagogical College, Dobrich, vol. 6, University Publishing House "K. Preslavski", 2013, pp. 124-129 (in Bulgarian).

Implementation of Binary Search Trees Via Smart Pointers

Ivaylo Donchev, Emilia Todorova
Department of Information Technologies, Faculty of Mathematics and Informatics
St Cyril and St Methodius University of Veliko Turnovo
Veliko Turnovo, Bulgaria

***Abstract*—Study of binary trees has prominent place in the training course of DSA (Data Structures and Algorithms). Their implementation in C++ however is traditionally difficult for students. To a large extent these difficulties are due not so much to the complexity of algorithms as to language complexity in terms of memory management by raw pointers – the programmer must consider too many details to ensure a reliable, efficient and secure implementation. Evolution of C++ regarded to automated resource management, as well as experience in implementation of linear lists by means of C++ 11/14 lead to an attempt to implement binary search trees (BST) via smart pointers as well. In the present paper, the authors share experience in this direction. Some conclusions about pedagogical aspects and effectiveness of the new classes, compared to traditional library containers and implementation with built-in pointers, are made.**

Keywords—abstract data structures; binary search trees; C++; smart pointers; teaching and learning

I. Introduction

From the C language, we know that pointers are important but are a source of trouble. One reason to use pointers is to have reference semantics outside the usual boundaries of scope [1]. However, it can be quite difficult to ensure that the life of a pointer and the life of the object to which it points will coincide, especially in cases where multiple pointers point to the same object. Such is the situation when an object must participate in multiple collections – each of them must provide a pointer to this object. To make everything correct it is necessary to be sure that:

- when destroying one of the pointers, take care that there are no dangling pointers or multiple deletions of the pointed object;
- when destroying the last reference to an object, to destroy the very object in order not to allow resource leaks;
- do not allow null-pointer dereference – a situation in which a null pointer is used as if it points to a valid object.

It is a must to have in mind such details to accomplish dynamic implementation of ADS (Abstract Data Structures) and often time for this exceeds time remaining to comment the structures and operations on them. Moreover, there are rare cases when these is a working implementation of a structure with carefully designed interface and methods written according to the best methodologies, but gaps can be identified in memory management only when a non-trivial situation occurs, such as copying large structures, transfer of items from one structure to another, or destruction of a large recursive structure. For each class representing ADS the programmer must also provide characteristic operations as well as correctly working copy and move semantics, exception handling, construction and destruction. This requires both time and expertise in programming at a lower level. The teacher will have to choose between emphasizing on language-specific features and quality of implementation or to compromise with them and to spend more time on algorithms and data structures. In an attempt to escape from this compromise, it is decided to change the content of CS2 course in DSA, include the study of smart pointers for resource management and with their help to simplify implementations of ADS to avoid explicit memory management which is widely recognized as error-prone [2].

In the work, the emphasis is on the implementation of linear structures (linked lists) and binary trees. This paper discusses only part of this work dedicated to binary search trees (BST).

The initial hypothesis is that a correct and effective implementation of BST is possible, which could relieve the work in two directions:

- operations with whole structures (trees): not having to implement copy and move semantics methods;
- shorter explanation and easier understanding of implementation of operations with elements of BST – include (insert element), search, delete.

The remaining content of the paper is as follows: Section II is a brief overview of language features for managing dynamic memory and its development. In paragraph III an implementation of Binary Search Trees (BST) is presented and compared to those based on build-in pointers. Section IV discusses effectiveness of the implemented structures and algorithms compared to the similar realization of the library container `std::set`. In section V some conclusions are made and recommendations are given for smart pointers usage in the DSA course.

II. Development of Language Features for Dynamic Memory Management

Before introducing of `new` and `delete` for work with dynamic memory, inherited from the C language functions

malloc, calloc, realloc and free are used, which are still available in C++ by including the header file <cstdlib>.

Memory blocks allocated by these functions are not necessarily compatible with those returned by new, so each must be handled with its own set of functions or operations. The problems with using these functions are related to unnecessary type conversions and error-prone size calculations (with sizeof).

Introduction of new and delete operators simplifies the syntax, but does not solve all problems. Especially in applications that manipulate complicated linked data structures it may be difficult to identify the last use of an object. Mistakes lead to either duplicate de-allocations and possible security holes, or memory leaks [2].

All the potential problems with locally defined naked pointers include:

- **leaked objects**: Memory allocation with new can cause (though rarely) an exception which is not handled. It is also possible the function execution to be terminated by another raised exception and the allocated with new memory to remain unreleased (it is not exceptions safety). Avoiding such resource leak usually requires that a function catch all exceptions. To handle deletion of the object properly in case of an exception, the code becomes complicated and cluttered. This is a bad programming style and should be avoided because it is also error prone. The situation is similar when the function execution is terminated by premature return statement based on some condition (early return);
- **premature deletion**: An object is deleted that has some other pointer to and later that other pointer is used.
- **double deletion**: There is a possibility to re-delete the object.

One way to circumvent these problems is to simply use a local variable instead of a pointer, but if we insist to use pointer semantics, the usual approach to overcome such problems is to use "smart pointers". Their "intelligence" is expressed in the fact that they "know" whether they are the last reference to the object and use this knowledge to destroy the object only when its "ultimate owner" is to be destroyed.

It is possible to consider that a "smart pointer" is RAII (Resource Acquisition Is Initialization) modeled class that manages dynamically allocated memory. It provides the same interfaces that ordinary pointers do (*, ->). During its construction it acquires ownership of a dynamic object in memory and deallocates that memory when goes out of scope. In this way, the programmer does not need to care himself for the management of dynamic memory.

For the first time standard C++98 introduces a single type of smart pointer – auto_ptr which provides specific and focused transfer-of-ownership semantics. auto_ptr is most charitably characterized as a valiant attempt to create a unique_ptr before C++ had move semantics. auto_ptr is now deprecated, and should not be used in new code. It works well in trivial situations – template auto_ptr holds a pointer to an object obtained via new and deletes that object when it itself is destroyed (such as when leaving block scope). Here auto_ptr is "smart" enough, but it appears that the problems entailed outweigh the benefit from it:

- copying and assignment among smart pointers transfers ownership of the manipulated object as well. That is, by default move assignment and move construction are carried out. Such is the situation with passing of auto_ptr as a parameter of the function. After function completes the memory allocated in the initialization of auto_ptr variable and then passed as argument to the function will be released (at destruction of the formal parameter) and will not be given back to this variable (the actual parameter). This will result in a dangling pointer. The auto_ptr provides semantics of strict ownership. auto_ptr owns the object that holds a pointer to. Copying auto_ptr copies the pointer and transfers ownership to the destination. If more than one auto_ptr owns the same object at the same time, program behavior is undefined.
- auto_ptr can not be used for an array of objects. When auto_ptr goes out of scope, delete runs on its associated memory block. This works for a single object, not for an array of objects that must be destroyed with delete [].
- because auto_ptr does not provide shared-ownership semantics, it can not even be used with Standard Library containers like vector, list, map.

Practice shows that to overcome (or at least limit) problems as described above it is not sufficient to use only one smart pointer class. Smart pointers can be smart in some aspects and carry out various priorities, as they have to pay the price for such intelligence [1], p. 76. Note that even now, with several types of smart pointers, their misuse is possible and it leads to wrong program behavior.

In the standard [3] instead of auto_ptr several different types of smart pointers are introduced (also called Resource Management Pointers) [4]. They model different aspects of resource management. The idea is not new – it formally originates from [5] and is originally implemented in the Boost library and only in 2011 became a part of the Standard Library. The basic, top-level and general-purpose smart pointers are unique_ptr and shared_ptr. They are defined in the header the file <memory>.

Unfortunately, excessive use of new (and pointers and references) seems to be an escalating problem. However, when pointer semantics is you really needed, unique_ptr is a very lightweight mechanism, with no additional costs compared to the correct use of built-in pointer [4], p. 113. The class unique_ptr is designed for pointers that implement the idea of exclusive (strict) ownership, what is intended auto_ptr to do. It ensures that at any given time only one smart pointer may point to the object. As a result, an object gets destroyed automatically when its unique_ptr gets destroyed. However,

transfer of ownership is permitted. This class is particularly useful for avoiding leak of resources such as missed `delete` calls for dynamic objects or when exception occurs while an object is being created. It has much the same interface as an ordinary pointer. Operator * dereferences the object to which it points, whereas operator -> provides access to a member if the object is an instance of a class or a structure. Unlike ordinary pointers, smart pointer arithmetic is not possible, but specialists consider this an advantage, because it is known that pointer arithmetic is a source of trouble. Use of `unique_ptr` includes passing free-store allocated objects in and out of functions (rely on move semantics to make return simple and efficient).

Copying or assignment between unique pointers is impossible if ordinary copy semantics is used. However, move semantics can be used. In that case, the constructor or assignment operator transfers ownership to another unique pointer.

Typical use of `unique_ptr` includes:

- ensuring safe use of dynamically allocated memory through the mechanism of exceptions (exception safety);
- transfer of ownership of dynamically allocated memory to function (via parameter);
- deallocating dynamically allocated memory for a function;
- storing pointers in the container.

A point of interest is the situation when `unique_ptr` is passed as a parameter of a function by rvalue reference, created by `std::move()`. In this case the parameter of the called function acquires ownership of `unique_ptr`. If this function then does not pass ownership again, the object will be destroyed at the completion of the function.

Using a unique pointer, as a member of a class may also be useful to avoid leak of resources. By using `unique_ptr`, instead of built-in pointer there is no need of a destructor because the object will be destroyed while destroying the member concerned. In addition, `unique_ptr` prevents leak of resources in case of exceptions which occur during initialization of objects – it is known that destructors are called only if any construction has been completed. So, if an exception occurs within a constructor, destructors will be executed for objects that have been already fully constructed. As a result there can be outflow of resources for classes with multiple raw pointers, if the first construction with `new` is successful, but the second fails.

Simultaneous access to an object from different points in the program can be provided through ordinary pointers and references, but it was already commented on the problems associated with their use. Often it is needed to make sure that when the last reference to an object is deleted, the object itself will be destroyed as well (which usually implies garbage collection operations – to deallocate memory and other resources).

`The shared_ptr` class implements the concept of shared ownership. Many smart pointers can point to the same object, and the object and its associated resources are released when the last reference is destroyed. The last owner is responsible for the destroying. To perform this task in more complex scenarios auxiliary classes `weak_ptr`, `bad_weak_ptr`, `enable_shared_from_this` are provided.

The class `shared_ptr` is similar to a pointer with a counter of the number of sharings (reference counter), which destroys the pointed object when this counter becomes zero. Imagine `shared_ptr` as a structure of two pointers – one to the object and one to the counter of sharings.

Shared pointer can be used as an ordinary pointer – to assign, copy and compare, to have access to the pointed object via the operations * and ->. A full range of copy and move constructions and assignments is available. Comparison operations are applied to stored pointers (usually the address of the owned object or `nullptr` if none). `shared_ptr` does not provide index operation. For `unique_ptr` a partial specialization for arrays is available that provides `[]` operator, along with * and ->. This is due to the fact that `unique_ptr` is optimized for efficiency and flexibility. Access to the elements of the owned by `shared_ptr` array can be provided through the indices of the internal stored pointer, encapsulated by `shared_ptr` (and accessible through the member function `get()`).

By using shared pointers the problems with dangling pointers can be avoided. This problem arises while pointers are stored in containers.

A problem with reference-counted smart pointers is that if there is a ring of objects that have smart pointers to each other, they keep each other "alive" – they will not be deleted even if no other objects are pointing to them from "outside" the ring. Such a situation often occurs in implementations of recursive data structures. C++11 includes a solution: "weak" smart pointers: these only "observe" an object but do not influence its lifetime. A ring of objects can point to each other with `weak_ptrs`, which point to the managed object but do not keep it in existence. Like raw pointers, weak pointers do not keep the pointed-to object "alive". The cycle problem is solved. However, unlike raw pointers, weak pointers "know" whether the pointed-to object is still there or not and can be interrogated about it, making them much more useful than a simple raw pointer would be.

In practice often happens a situation when the programmer hesitates which version of a smart pointer to use – `unique_ptr` or `shared_ptr`. The advice is to prefer `unique_ptr` by default, because later move-convert to `shared_ptr` can be done if needed. There are three main reasons for this [6]:

- try to use the simplest semantics that are sufficient;
- a `unique_ptr` is more efficient than a `shared_ptr`. A `unique_ptr` does not need to maintain reference count information and a control block under the covers, and is designed to be just about as cheap to move and use as a raw pointer;
- starting with `unique_ptr` is more flexible and keeps the options open.

In this particular case, however, it is necessary to start from the very beginning with `shared_ptr`, because being recursive by definition, binary trees that have to be implemented with smart pointers, and this cannot do without shared ownership.

III. Implementation of Binary Search Trees

Most attention in the course is given to binary search trees, so here the focus is only on the implementation. The traditional implementation interface with build-in pointers looks like this:

```
template <typename T>
class BTree {
   struct Node {
        T key;
        Node* left;
        Node* right;
        Node();
        Node(T);
   };
typedef Node* pNode;  //pNode& instead of Node*&
   pNode root;
   //........ some helper functions here ........
public:
        BTree() : root(nullptr){}
        ~BTree();
        BTree(const BTree&);
        BTree(BTree&&);
        BTree& operator =(const BTree&);
        BTree& operator =(BTree&&);
        bool insert(T);
        bool remove(T);
        void inorder(void(*)(pNode&));
        void preorder(void(*)(pNode&));
        void postorder(void(*)(pNode&));
        void breath_first(void(*)(pNode&));
        size_t height();
        Node* find(T);
};
```

Beside the special member functions methods are added to insert, search and remove elements, and various deep-first (`inorder`, `preorder`, `postorder`) and breath-first traversals. A number of additional functions are included. Their implementation is a question of interest, for example, calculating the height of the tree and, if there is enough time, balancing. For implementation of these operations, recursive algorithms are preferred because they are shorter and more intuitive. Most difficulties are met with the deletion, which is normal – the algorithm is most complex.

Since the aim is to count on the reliability, in the course it is chosen to follow the methodology for verification of object-oriented programs as proposed in [7].

In order to simplify the technical part and to focus on algorithms, implementing the operations on trees from 2013-2014, it is decided to choose implementation with smart pointers. The initial expectation is that it is possible to avoid all methods of copy and move semantics, destructors for nodes and whole trees.

The interface of smart pointers implementations with which the work is started is the following:

```
template <typename T>
class Tree {
  struct Node {
         T key;
         shared_ptr<Node> left;
         shared_ptr<Node> right;
         Node():key(), left(), right(){}
         Node(T x):key(x),left(), right(){}
  };
  shared_ptr<Node> root;
//...
public:
  Tree():root(){}
  ~Tree();
  Tree(Tree&&) = default;
  Tree& operator =(Tree&&) = default;
  Tree(const Tree&);
  Tree& operator =(const Tree&);
  bool push(T);
  bool remove(T);
  void inorder();
  shared_ptr<Node> find(T x) {
         return find(x, root);
  }
  void breath_first();
  size_t height(){
         return height(root);
  }
};
```

Because of recursive algorithms that are used for each operation two functions had to be written – one private, with additional parameter the node from which to start. So public method is very short and just calls the corresponding private method that implements the algorithm. For example the public method for deleting:

```
template <typename T>
bool Tree<T>::remove(T x){
         return remove(x, root);
}
```

calls the private method `remove(T, shared_ptr<Node>&)` where the second parameter is the root of the tree:

```
template <typename T>
bool Tree<T>::remove(T x, shared_ptr<Node>& p) {
   if(p && x < p->key)
        return remove(x, p->left);
   else if(p && x > p->key)
        return remove(x, p->right);
        else if(p && p->key == x) {
           if(!p->left)
                p = p->right;
           else if(!p->right) p = p->left;
                else {
        shared_ptr<Node> q = p->left;
        while(q->right) q=q->right;
        p->key = q->key;
        remove(q->key, p->left);
        }
        return true; }
        return false;}
```

We note that the code for this method is 37% shorter than the code for the corresponding raw pointers implementation (due mainly to the fact that there is no need to call `delete`). In addition readability of code is improved. For inserting a node there is no difference between the amounts of code – both methods have 16 rows.

For educational purposes, all operations with a single tree run normally, but when a larger tree is tested, a "stack overflow" error appears during automatic tree destruction at the end of the program. With a standard size of 1 MB stack error occurs even for destruction of a tree of 29,000 integers. Because of recursive links, a situation arises where one node keeps "alive" the whole structure. This on one hand requires a large stack, and on the other – can lead to significant delays in demolition of the structure. So the choice is to add a destructor, instead of increasing stack size from the settings of the linker. The decision is not to work for efficiency and chose the easiest option – using the method for deletion. As such, the destructor looks like this:

```
~Tree(){
        while (root) remove(root->key);
}
```

As for the implementation of special member functions, defaulting of move constructor and move assignment operator works and it is not needed to implement move semantics, but copy semantics requires to write appropriate methods, because it is needed to copy the entire tree structure, so as to obtain a true copy of the tree, not just tree, which contains the same elements.

Comparing the overall implementation of trees with raw pointers, the conclusion is that smart pointers give short and easy to understand code without apparent loss of efficiency (Table 1).

IV. Performance Evaluation

In order to evaluate the efficiency of smart pointers implementation an experiment is carried out in which times for typical operations with binary trees, implemented with and without smart pointers, are compared.

Three conversions are compared: traditional row pointer implementation, new smart pointer and library implementation `std::set` (Table 1). Note that `std::set` is typically implemented in libraries as a red-black tree. This may adversely affect time for generating the tree (for coloring and balance), but improves search speed.

The same data is used in the experiment: 100,000 randomly generated unique strings of length of 20 stored in a text file. They are used to construct trees. The first operation "Add element" reads all strings from the file and stores them in the relevant tree. For each tree, the text file is opened and read again. For unbalanced versions, a tree with height of 38 is obtained.

In testing for search and remove elements another file is used, which records 10,000 strings that are found in the tree. The algorithm makes search and remove operations for exactly these elements.

TABLE I. Test Results for Binary Trees

Operations	*Binary Search Tree Implementations*		
	Row Pointers	Smart Pointers	`std::set`
Add element	438	453	156
Search	31	32	15
Remove	47	46	32

Note: time in milliseconds

The results show that there is practically no difference in performance between implementation of operations with build-in and smart pointers, which is a good argument to continue to study smart pointers in the course DSA. Some surprise is the time for `std::set` in operations creating structure (adding operation), which is three times better. Apparently, extra time for coloring and balancing the tree is offset by the lower height of the red-black tree – `std::set` for these input data theoretically the tree can get a height of 12, and as mentioned before, the tree in our implementations has height 38. For the same reasons, search time in our implementations is 2 times worse, and time for removing elements – 1.5 times worse library implementation.

V. Conclusion

The initial hypothesis regarding the implementation of BSTs with smart pointers is proven partially. It is not possible to do the work entirely without implementation of methods of copy and move semantics, but their code turns out to be short, clear and easily understandable for students. Moreover, move semantics can be provided by defaulted move constructors and assignment operators. It is considered that the second part of the hypothesis, namely the shorter and clearer implementation of basic operations with data structures is fully achieved. In addition, smart pointer versions do not require user-defined exception handling.

Since there is not enough empirical data, the advantage of this way of teaching DSA cannot be proved yet, but even without holding a strictly formal pedagogical experiment, it can be stated that results of students tests, homework and exams are comparable to those demonstrated by their colleagues trained in previous years under the old program.

Implementation of ADS with smart pointers is more clear and concise, but requires spending time to study in addition templates and essential elements of the STL, though not in detail. This could be facilitated by reorganizing CS1 course Programming Fundamentals, where to underlie learning C++11/14 and STL. Note that for the presented implementations it is not needed even to know the full interface for work with smart pointers. In most situations the interface of build-in pointers is sufficient plus function `make_shared` and possibly member function `reset`. While working with students during the school year some difficulties are met in debugging of programs related to discovery of logical errors in memory management, most often connected with its release.

It is appropriate to add an intermediate output (operator `cout`) in destructors as of DSA, as of the elements held in them (if they are of user-defined types). In this way, it is easy to detect situations where objects remain undestroyed.

Regarding the applicability of smart pointers in actual programming the opinion of Stroustrup should be mentioned, that they "are still conceptually pointers and therefore only my second choice for resource management – after containers and other types that manage their resources at a higher conceptual level" [4], p. 114. The results of comparative tests also show that library containers are sufficiently effective. In order to learn smart pointers it is necessary to get into STL. On one hand, it is better to teach students how to use its efficient and reliable containers. On the other hand though, as future professionals they must be able to independently implement such containers – to develop creative thinking. It is therefore not a bad idea to do so with smart pointers as well – one more opportunity provided by the STL.

REFERENCES

[1] Josuttis, N. M. (2012). *The C++ Standard Library: A Tutorial and Reference*. Addison-Wesley Professional; 2nd edition (April 9, 2012).

[2] Boehm, H. & Spertus, M. (2009). *Garbage Collection in the Next C++ Standard*. Proceedings of the 2009 international symposium on Memory management, pp. 30-38. ACM New York. doi>10.1145/1542431.1542437

[3] ISO/IEC. (2011). *International Standard ISO/IEC 14882:2011(E) Information technology – Programming languages – C++* (3rd ed.)

[4] Stroustrup, Bj. (2013). *The C++ Programming Language, 4th Edition*. Addison-Wesley Professional; 4th edition (May 19, 2013)

[5] Dimov, P., Dawes, B. & Colvin, G. (2003). *A Proposal to Add General Purpose Smart Pointers to the Library Technical Report*. C++ Standards Committee Papers. Document number: N1450=03-0033 http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1450.html

[6] Sutter, H. (2013). *Sutter's Mill. GotW #89 Solution: Smart Pointers*. http://herbsutter.com/2013/05/29/gotw-89-solution-smart-pointers/

[7] Todorova, M., Kanev, K. (2012). *Educational framework for verification of object-oriented programs*, in Proceedings of the 2012 Joint International Conference on Human-Centered Computer Environments, ACM, New York, pp. 23-27

Revised Use Case Point (Re-UCP) Model for Software Effort Estimation

Mudasir Manzoor Kirmani
Research Scholar, School of CS & IT
Maulana Azad National Urdu University
Hyderabad, India

Abdul Wahid
Dean and Head School of CS & IT
Maulana Azad National Urdu University
Hyderabad, India

***Abstract*—At present the most challenging issue that the software development industry encounters is less efficient management of software development budget projections. This problem has put the modern day software development companies in a situation wherein they are dealing with improper requirement engineering, ambiguous resource elicitation, uncertain cost and effort estimation. The most indispensable and inevitable aspect of any software development company is to form a counter mechanism to deal with the problems which leads to chaos. An emphatic combative domain to deal with this problem is to schedule the whole development process to undergo proper and efficient estimation process, wherein the estimation of all the resources can be made well in advance in order to check whether the conceived project is feasible and within the resources available. The basic building block in any object oriented design is Use Case diagrams which are prepared in early stages of design after clearly understanding the requirements. Use Case Diagrams are considered to be useful for approximating estimates for software development project. This research work gives detailed overview of Re-UCP (revised use case point) method of effort estimation for software projects. The Re-UCP method is a modified approach which is based on UCP method of effort estimation. In this research study 14 projects were subjected to estimate efforts using Re-UCP method and the results were compared with UCP and e-UCP models. The comparison of 14 projects shows that Re-UCP has significantly outperformed the existing UCP and e-UCP effort estimation techniques.**

***Keywords*—Use Case Point; Extended Use case point; Revised Use case Point; Software Effort Estimation**

I. INTRODUCTION

Software being indispensable and inevitable entity which is currently ruling almost all the modern day operability directly or indirectly having crucial attributes associated with it and failure of those can prove out to be of grave damage to different Industrial and societal parameters. Software cost estimation is one of the pivotal issues in modern software development industry making it the most important activity in software engineering and software project management domain [1]. Effort estimation is an activity to estimate the number of business activities of workers as well as how long it takes to accomplish a software development project. The effort estimation activity is very important to know how much relevant value of software is generated within the specified parameters. Accurate and reliable software development effort estimates have always been encouraging for project managers [10]. There are number of methods, tools and techniques which can be put into practice to estimate the cost of the software ranging from traditional modelling to modern day modelling.

Based on the literature available the different models like analogy based model, experience based model, LOC, KLOC, COCOMO and Function points, have played vital role in estimating effort of software development projects [4]. However, the requirement engineering spawned to higher levels of complexity these methods had to counter the challenging task of performing at higher levels of acceptability and scalability [6] [7]. To overcome this Use Case Points (UCP) were introduced to estimate the effort of the software development project in early stage of development [11] [18].

Most of the software development organizations are using Object-Oriented technology based approach for developing software. The basic building block of an object oriented design is Use Case diagram which is prepared in the early stages of design after clearly understanding the requirement [9]. A use case diagram is the simplest representation of a user's interaction with the system. These diagrams can portray different types of users and their interaction pattern with the system. Use Case Diagrams are considered to be useful for approximating early estimation of a software project. The use case point model for effort estimation was first proposed by Gustav Karner in 1993 [14], which was focused to predict the total amount of resources required for developing a software system with object-oriented technology in the early stages of software development process.

This new use case point method performed well in comparison to the other techniques in practice and has also gained wide popularity [3] [5] [12] [13] [24]. Researchers from academia as well as industry have shown interest in the Use Case point based approaches because of the promising results obtained along with their applicability in early steps of software development [17]. There have been a number of approaches proposed in the literature [12] [15] [16] [22] [23] [24]. However, there is no criteria that could be used to aid practitioners in selecting appropriate approaches which is suitable for estimation of efforts for different software development projects. Even though UCPs have played a challenging role in software effort estimation, further enhancement is needed in some of its corresponding parameters to ensure further improvement in bridging the gap between the actual and estimated efforts [1] [2] [19] [20].

This research work is an effort to propose a refined model based on UCP and e-UCP methods in order to improve the

efficiency of effort estimates for software development projects in the early stages of development. The rest of research paper gives an overview about UCP and e-UCP models in section II & III respectively. Section IV Re-UCP model for estimation of effort for software development projects is proposed with refinement in the complexity of actors and use cases. In section V experimental results of 14 projects are presented and discussed by highlighting the effectiveness of the proposed model. The conclusion and the proposal of future works are given in Section VI

II. Use Case Points Method

The UCP method is the extension of Function Point method with the benefit of requirement analysis in object-oriented process. It starts with measuring the functionality of the system based on the use case model in a count called Unadjusted Use Case Point (UUCP). The same technical factors are used as of Function Points. The UCPs shows an estimation of the size of the system which can be further mapped to man hours in order to calculate the effort required to develop a system.

Actors and use cases are classified into simple, average and complex categories according to their complexity and is assigned some weight factor. An actor is defined as "Simple", if it interacts with the system with the help of a defined application programming interface (API). An actor is defined as "Average", if it interacts with the help of an Interactive or Protocol-Driven Interface. The actor is defined as "Complex", if it interacts through a Graphical User Interface. The assigned weight factor for simple, average and complex are 1, 2 and 3 respectively.

Similarly use case is defined as "simple", if the number of transaction is less than 3, "average", if the number of transaction is between 4 and 7 and "complex", if the number of transaction is more than 7. The assigned weight factors for simple, average and complex are 5, 10 and 15 respectively.

After calculating UUCP, the use case points are calculated by multiplying UUCP to technical complexity factor (TCF) and Environmental factors (EF). The TCF corresponds of 13 different parameters and ECF corresponds of 8 parameters.

UUCP = UAW + UUCW

UCP = UUCP * TCF * EF

Further the effort is estimated by mapping the UCP with man-hours.

Effort = UCP * PHper UCP

Where PHper UCP is Person Hours per UCP.

III. Extended Use Case Point Method (E-Ucp)

The extended use case point method (e-UCP) is a revised version of UCP method and was proposed by Kasi Perivasamy and Aditi Ghade in 2009 [21]. The e-UCP model considers some additional information about the relationships between actors and use cases. The e-UCP is focussed on internal details of a use case by including the use case narrative in effort estimation process of a software development project in the early stages of development. It starts with measuring the functionality of the system based on the use case model in a count called Unadjusted Use Case Point (UUCP). The technical factors and environmental factors used were similar to UCP method of effort estimation. The e-UCPs shows an estimation of the size of the system which can further be mapped to man-hours in order to calculate the effort required to develop the system.

The categorization of actors was modified in e-UCP method of software effort estimation and the number of actor categories was increased from 3 to 7. The modified categories of actors were 'very-simple', 'simple', 'less-average', 'average', 'complex', 'more-complex', 'most-complex' and the corresponding weight assigned were 0.5, 1.0, 1.5, 2.0, 2.5, 3.0 and 3.5 respectively [21]. All the assigned values from 'very simple' to 'most complex' were multiplied by their corresponding weight factor and the summation of all calculated values is the actor weight.

Similarly the categorization was modified by increasing the number of use cases from 3 used in UCP to 4 in e-UCP [21]. The modified categories of use cases were 'simple', 'average', 'complex' and 'most complex' and the corresponding weight assigned was 0.5, 1.0, 2.0 and 3.0 respectively [21]. All the assigned values from 'simple' to 'most complex' were multiplied by their corresponding weight factor and the summation of all calculated values is the value of use case weight.

A new adjustment factor named as 'use case narrative' was added in extended use case point method of software effort estimation (e-UCP). The classified parameters for use case narrative were 'input-parameter', 'output-parameter', 'a-predict-in-precondition', 'a-predict-in-post-condition', 'an-action-in-successful-scenario' and 'an-exception' and the corresponding parameter weight were 0.1, 0.1, 0.1, 0.1, 0.2 and 0.1 respectively [21]. All the assigned values for different use case narrative parameters were multiplied by their corresponding weight factor and the summation of all calculated values is the value of use case narrative weight.

The calculated value of actor weight, use case weight and use case narrative weight was used in the following equation to calculate unadjusted use case points (UUCP).

Unadjusted Use Case Points (UUCP)

UUCP = Use Case Weight + Actor Weight + Narratives Weight

e-UCP = UUCP * TCF * EF

where TCF -Technical Complexity Factor

EF - Environmental Factor.

The number of parameters in TCF technical complexity factor and EF environmental factor used in e-UCP were same as in case of UCP.

Effort = e-UCP * PHper UCP

where PHper is Person Hours per UCP.

IV. Revised Use Case Point Method (Re-Ucp)

Re-UCP is an extension to UCP and e-UCP model wherein all the existing behaviour and implementation parameters of

the two models are pondered comprehensively in order to design the generic framework for software effort estimation which can have adaptable behaviour for all range of projects with varying level of complexity and have the futuristic scope of scalability which can handle the agile software development activities. In Re-UCP model the functionality of the system is measured by calculating all the use case points in the system. The functionality of the system is estimated by the collective impact of corresponding factors associated with actors of the system, behaviour of use case, impact of environment and role of technical factors over the use case point.

In Re-UCP, actors, use cases, environmental and other technical factors are further categorised to associate a particular impact factor on the specific use case activity rather than functionality of the system. Unlike UCP or e-UCP wherein either actor or use case is divided into simple, average, or complex with some weighting strategy, Re-UCP uses different categorization of actor and use cases. Actor in Re-UCP are categorised into simple, average, complex and critical with weight parameters of 1, 2, 3 and 4 respectively. Similarly use cases are also divided into simple, average, complex and critical categories and the detailed description of actors and use cases with their corresponding weighting factor is given in table [I].

TABLE I. ACTOR TYPE AND RESPECTIVE WEIGHT

Actor Type	Weight
Simple	1
Average	2
Complex	3
Critical	4

An actor is defined as "Simple", if it interacts with the system with the help of a defined application programming interface (API). An actor is defined as "Average", if it interacts with the help of an Interactive or Protocol-Driven Interface. The actor is defined as "Complex", if it interacts through a Graphical User Interface. The actor is defined as "Critical", if it interacts with modules wherein real time action is taken or complexity is very high. The weight parameters for simple, average, complex and critical are 1, 2, 3 and 4 respectively.

Similarly use case type is defined as "Simple", if the number of transaction is less than or equal to 4, "Average", if the number of transaction is between 5 and 8 , "Complex", if the number of transaction is between 9 and 15 and "Critical", if the number of transactions is greater than 15. The assigned weight factors for simple, average, complex and critical are 5, 10, 15 and 20 respectively and the same is given in table [II].

TABLE II. USE CASE TYPE AND RESPECTIVE WEIGHT

Use Case Type	No. of Transactions	Weight
Simple	<=4	5
Average	5 to 8	10
Complex	9 to 15	15
Critical	>15	20

Total Actor and Use case weight is calculated as:

UAW ---- Unadjusted Actor Weight

UAW =Σ (No. of actors * their respective weight factors) -------- (1)

UUUW ---- Unadjusted Use Case Weight

UUCW = Σ (No. of Use cases * their respective weight factors) -------- (2)

UUCP ---- Unadjusted Use Case Points

UUCP = UAW+UUCW ---------- (3)

The revised use case points are calculated as

Re-UCP = UUCP * TCF * ECF ---------- (4)

Where TCF-Technical Complexity Factor

ECF-Environmental Complexity Factor

In order to estimate the overall use case points of the system some other factors corresponding to development environmental and technical parameters need to be considered in development process and are called as Technical Complexity Factor (TCF) and Environmental Complexity Factor (ECF) respectively.

In the UCP and e-UCP models of effort estimation the TCF (Technical Complexity Factor) correspond of 13 different parameters which were assigned value from range 0 up to 5. The value '0' implies that the parameter is irrelevant and the assigned value will increase with the increase in significance. However, if the value is '5' then the significance of the corresponding parameter is treated as essential. In Re-UCP the number of parameters in TCF has been increased from 13 to 14 wherein scalability parameter was included as 14th parameter. The label for the 14th parameter in TCF is given as T14 and the value assigned for the parameter will be 0 in case of irrelevant up to 5 in case of essential.

Scalability can be defined as the ability of the system to handle increased workloads without adding resources to the existing system by repeatedly applying a cost-effective strategy for extending system capacity. By the progress of time the level of administering the projects varies with good levels of complexity. To make our system to be more adaptable to handle such dynamic projects is very much indispensable activity in present scenario and hence the scalability factor in Re-UCP has been incorporated to address this issue.

TCF is one of the factors used to integrate the predominance of the various enlisted technical factors of the system on the overall developmental process and simultaneously estimating the effect of impact on the overall software effort estimation process. All the technical factors from T1 to T14 are multiplied by their corresponding weight factor as described in the table [III] and the summation of all calculated values is the calculated value of technical complexity factor (TCF).

TABLE III. TECHNICAL COMPLEXITY FACTOR AND WEIGHT

Factor	Description	Weight
T1	Distributed system	2
T2	Response or throughput performance objectives	1
T3	End-user efficiency (online)	1
T4	Complex internal processing	1
T5	Code must be reusable	1
T6	Easy to install	0.5
T7	Easy to use	0.5
T8	Portable	2
T9	Easy to change	1
T10	Concurrent	1
T11	Includes special security features	1
T12	Provides direct access for third parties	1
T13	Special user training facilities are required	1
T14	Scalability	2

TF subsequently is used to obtain the value of the Technical Complexity Factor (TCF).

$$TCF = 0.6 + (0.01 * \sum_{i=1}^{14} TF_i) \quad \text{---------- (5)}$$

In the UCP and e-UCP models of software effort estimation the ECF (Environmental Complexity Factor) corresponded of eight parameters and were labelled from E1 to E8. Each parameter of ECF was assigned a value from the range 0 up to 5 where '0' implied that the developed had no experience of the corresponding parameter and if the developer experience was better than a higher value was assigned from the range. However, if the assigned value of the parameter was '5' then the developer was considered as expert. In Re-UCP the number of parameters in ECF was increased from 08 to 09. The ninth parameter included was project methodology which describes the experience of the developer in the project methodology selected for the development of the software project. The label for the ninth parameter in ECF is given as E9 and the value assigned for the parameter will be 0 in case of inexperienced developer up to 5 in case of expert developer.

All the environmental complexity factors from E1 to E9 are multiplied by their corresponding weight factor as described in the table [IV] and the summation of all calculated values is the value of environmental complexity factor (ECF).

TABLE IV. ENVIRONMENTAL FACTOR AND WEIGHT

Factor	Description	Weight
E1	Familiarity with the project	1.5
E2	Application Experience	0.5
E3	OO Programming Experience	1.0
E4	Lead Analyst Capability	0.5
E5	Motivation	1.0
E6	Stable requirements	2.0
E7	Part Time Staff	-1.0
E8	Difficult Programming Language	-1.0
E9	Project Methodology	1.0

EF value is used to obtain the Environmental Complexity Factor (ECF).

$$ECF = 1.4 + (-0.03 * \sum_{i=1}^{9} EF_i) \quad \text{-------- (6)}$$

The total number of all the revised use case points is calculated by multiplying UUCP, TCF and ECF

Re-UCP = UUCP * TCF * ECF---(from eq. 4)

The value of TCF is calculated using equation 5 and the value of ECF using equation 6. Both the values of TCF and ECF are multiplied with UUCP to calculate the number of revised use case points (Re-UCP). The efforts per Re-UCP is calculated by using 20 man-hours per UCP as was suggested by Karnar [14]. The equation (7) is used to convert number of Re-UCP into person hours and the same is given below

Effort = UCP * PHper UCP ----------(7)

Where PHper UCP is Person Hours per UCP

V. RESULTS AND DISCUSSION

A group of nearly 51 students selected from both UG and PG course were trained for a week long time to get hands-on the experience of using UCP, e-UCP and newly designed method Re-UCP. After the completion of training the students were divided into 11 groups wherein 5 groups with 5 members each, 5 groups with 4 members each and 2 groups with 3 members each. 14 different software projects with varying complexity were given to these 11 groups with some groups working on maximum of two projects. All the groups were subjected to use UCP, e-UCP and Re-UCP models for estimation software effort for 14 projects. The resultant data reporting format was standardized for all projects wherein all groups were asked to document use case points, using UCP, e-UCP and Re-UCP separately. The resultant data is given in the table [V] and the behaviour of this trend is represented graphically in fig. 1.

TABLE V. ESTIMATING THE TOTAL NUMBER OF USE CASE POINTS IN UCP, E-UCP AND RE-UCP

Project Name	use case points in UCP model	use case points in e-UCP model	use case points in re-UCP model
P1	128.0	134.9	138.7
P2	342.6	304.7	291.3
P3	253.4	279.8	282.8
P4	073.6	090.3	108.9
P5	096.3	099.1	122.7
P6	115.0	117.2	114.9
P7	276.4	250.4	260.3
P8	169.4	161.6	156.4
P9	067.7	065.8	063.7
P10	121.4	110.7	105.2
P11	228.2	210.9	199.8
P12	187.3	185.0	183.0
P13	208.6	210.6	192.0
P14	189.7	190.0	169.6

After analysing the bars from project 1 up to project 14 as given in fig. 1, in most of the projects the number of use cases points in case of Re-UCP is lesser when compared with UCP and e-UCP.

The use case points were calculated and then accordingly the total efforts estimation were carried out by multiplying number use case point with productivity factor of 20 man-hour as recommended by Karner [14]. The results obtained are given in table [VI] and the behaviour of all the three models is shown in fig. 2.

TABLE VI. EFFORT ESTIMATION BY USING UCP, E-UCP AND RE-UCP

Project Name	Actual Efforts	UCP Estimated Effort	e-UCP Estimated Effort	Re-UCP Estimated Effort
P1	2890	2560	2698	2774
P2	5600	6852	6094	5826
P3	5760	5068	5596	5656
P4	1925	1472	1806	2178
P5	2175	1926	1982	2454
P6	2180	2300	2344	2298
P7	4230	5528	5008	5206
P8	2870	3388	3232	3128
P9	1190	1354	1316	1274
P10	1930	2428	2214	2104
P11	3880	4564	4218	3996
P12	3350	3746	3700	3660
P13	3290	4172	4212	3840
P14	3080	3794	3800	3392

The pattern shown in fig. 2 gives an overview of estimated efforts in person hours using UCP, eUCP and Re0UCP. However, the observations from the table[vi] clearly indicate that the estimated effort using Re-UCP methods are more closed to actual efforts in comparison with estimated effort using UCP & e-UCP methods.

The deviation which is calculated by subtracting actual efforts from estimated efforts was calculated. The value of deviation can either be positive or negative and positive deviation explains that the estimated effort us greater than actual effort whereas negative deviation explained that the estimation effort is lesser than actual effort. The results of calculation for deviation of UCP, e-UCP and Re-UCP against the actual estimated effort is given in table [VII].

TABLE VII. DEVIATION OF UCP, E-UCP AND RE-UCP FROM THE ACTUAL ESTIMATED EFFORT

Project Name	Actual Efforts	UCP Deviation	e-UCP Deviation	Re-UCP Deviation
P1	2890	-330	-192	-116
P2	5600	1252	494	226
P3	5760	-692	-164	-104
P4	1925	-453	-119	253
P5	2175	-249	-193	279
P6	2180	120	164	118
P7	4230	1298	778	976
P8	2870	518	362	258
P9	1190	164	126	84
P10	1930	498	284	174
P11	3880	684	338	116
P12	3350	396	350	310
P13	3290	882	922	550
P14	3080	714	720	312

The graphical representation of the calculated deviation for UCP, e-UCP and Re-UCP against the actual effort estimation is given in fig. 3. The results represented graphically in fig. 3 shows that the project P1, P3, P4 and P5 have negative deviation whereas projects P2, P6 through P14 have positive deviation. In most of the cases across all projects from p1 to p14 there is very less deviation either positive or negative calculated using Re-UCP software effort estimation methods when compared to UCP and e-UCP method of effort estimation.

The proposed model further showed that among the methods compared (the UCP, e-UCP and Re-UCP, effort and deviation of the effort estimation methods) the performance of Re-UCP in comparison to UCP and e-UCP has improved. In project P1 and P3 the estimated effort is greater than the efforts obtained by either of the three effort estimation methods. In projects P2, P6 to P14 the estimated effort is less than the efforts acquired by either of these specified methods. Among the fourteen projects the efforts obtained in nine projects by Re-UCP is less than the efforts estimated using UCP and e-UCP methods. In projects P1, P3, P6, P8 to P14 the estimated deviation by using Re-UCP method is very less as compared to UCP and e-UCP.

VI. CONCLUSION

The effort estimation for any software development project should be carried out in the early stages of development in order to reduce the gap between the estimated effort and actual effort. To cope with the effort estimation of the projects with varying complexities Re-UCP method is proposed to cater needs of estimating effort in early stages of software development. The actors and use cases were categorized in four categories as simple, average, complex and critical. Scalability parameter was incorporated in TCF as the fourteenth parameter and Project Methodology was introduced in ECF as the ninth parameter. The performance of revised use case point model (Re-UCP) has shown improvements in estimating the efforts for software development projects with minimum trends in deviation from the actual efforts when compared with UCP and e-UCP effort estimation methods on 14 projects carried out by 11 groups after receiving proper training in the beginning.

Proposal for Future Work

The comparison of Re-UCP, UCP and e-UCP estimates, needs to be tested with data from successfully completed projects, from international and national software estimation data store organizations. The data from software development organization can be used as well to further test the performance of estimates using Re-UCP, UCP and e-UCP methods of software effort estimation. Therefore, future research in this domain needs to be carried out to strengthen the approaches available for software effort estimation which will help the developers in reducing the gap between actual efforts and estimated efforts.

REFRENCES

[1] Alves, R., Pedro V., and Nuno J. N., 2013. Improving software effort estimation with human-centric models: a comparison of UCP and iUCP accuracy. Proceedings of the 5th ACM SIGCHI symposium on Engineering interactive computing systems. ACM.

[2] Anda B., Endre A., and Kirsten R., 2002. Improving estimation practices by applying use case models. Product Focused Software Process Improvement. Springer Berlin Heidelberg, 383-397.

[3] Apol P. S., Sholiq and Ningrum P. A., 2014. Critical Review of the effort rate value in use case point method for estimating software development effort. Journal of Theoretical and Applied Information Technology. 59(3):735-744.

[4] Ashman, R., 2004. Project estimation: a simple use-case-based model. IT professional. 6(4): 40-44.

[5] Azzeh, M., 2013. Software cost estimation based on use case points for global software development. 5th International Conference on Computer Science and Information Technology (CSIT), IEEE. 214-218.

[6] Carroll, E. R., 2005. Estimating software based on use case points. In Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, ACM. 257-265.

[7] Clemmons, R. K., 2006. Project estimation with use case points. The Journal of Defense Software Engineering. 18-22.

[8] Eberendu, A. C., 2014. Software Project Cost Estimation: Issues, Problems and Possible Solutions. International Journal of Engineering Science Invention 3(II): 38-43.

[9] Issa, A., Odeh, M., and Coward, D., 2007. Can Function Points be Mapped to Object Points. The International Arab Journal of Information Technology. 4(1): 41-49.

[10] Lynch J., and Chaos M., October 2009. The Standish Group. Boston. Available online: http://www.standishgroup.com/ newsroom/ chaos_2009.php.

[11] Jena, P. P., and Mishra, S., 2014. Survey Report on Software Cost Estimation using Use Case Point Method. International Journal of Computer Science & Engineering Technology. 5(04): 280–287.

[12] Jha, P., Jena, P. P., and Malu, R. K., 2014. Estimating Software Development Effort using UML Use Case Point (UCP) Method with a Modified set of Environmental Factors. International Journal of Computer Science and Information Technologies (IJCSIT). 5(3): 2742-2744.

[13] Jones, T. C. , Estimating software costs. McGraw-Hill, Inc. 1998.

[14] Karner, G., 1993. Metrics for objectory. Sweden: University of Linköping. Sweden. No. LiTHIDA- Ex-9344:21.

[15] Kumari, S., and Pushkar, S., 2013. Performance Analysis of the Software Cost Estimation Methods: A Review. International Journal of Advanced Research in Computer Science and Software Engineering. 3(7): 229-238.

[16] Kumari, S., and Pushkar S., 2013. Comparison and Analysis of Different Software Cost Estimation Methods. International Journal of Advanced Computer Science and Application. 4(1).

[17] Mohagheghi, Parastoo, Bente A., and Reidar C., 2005. Effort estimation of use cases for incremental large-scale software development. Proceedings. 27th International Conference on Software Engineering, ICSE 2005, IEEE.

[18] Nagar, C., 2011. Software efforts estimation using Use Case Point approach by increasing Technical Complexity and Experience Factors. International Journal of Computer Science and Engineering (IJCSE). 3(10): 3337-3345

[19] Nassif, A. B., 2010. Enhancing Use Case Points Estimation Method using Soft Computing Techniques. Journal of Global Research in Computer Science. 1(4): 12-21.

[20] Ochodek M., Nawrocki, J., and Kwarciak, K., 2011. Simplifying Effort Estimation based on Use Case Points. Sciencedirect, Elsevvier. 200-213.

[21] Periyasamy, K., and Ghode, A., 2009. Cost estimation using extended use case point (e-UCP) model. International Conference on Computational Intelligence and Software Engineering, CiSE 2009 IEEE. 1-5.

[22] Robiolo, G., and Orosco, R. 2008. Employing use cases to early estimate effort with simpler metrics. Innovations in Systems and Software Engineering. 4(1): 31-43.

[23] Ruhe, M., Jeffery, R., and Wieczorek, I., 2003. Cost estimation for web applications. Proceedings. 25th International Conference on Software Engineering, IEEE. 285-294.

[24] Schneider, G., and Winters, J., 1998 Applying Use Cases – A Practical Guide.Addison-Wesley.

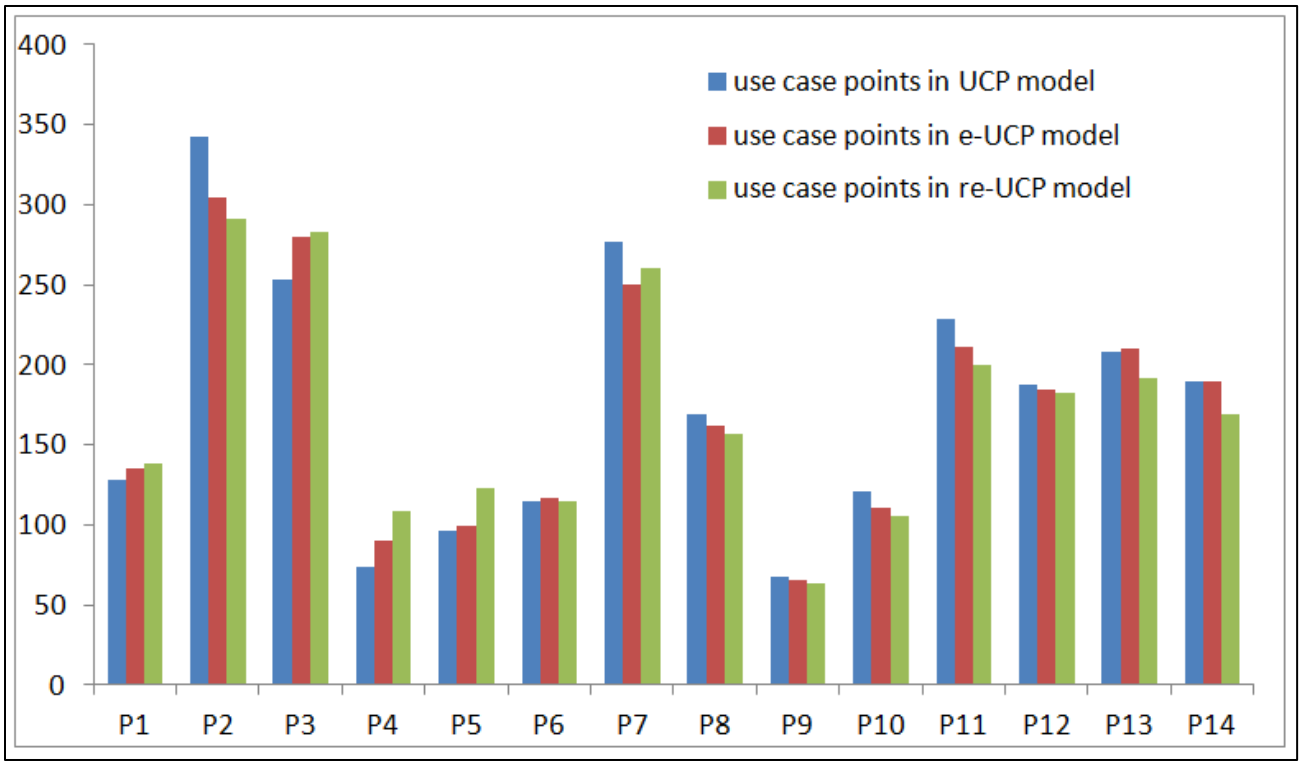


Fig. 1. Number of Use Case Points using UCP, e-UCP and Re-UCP

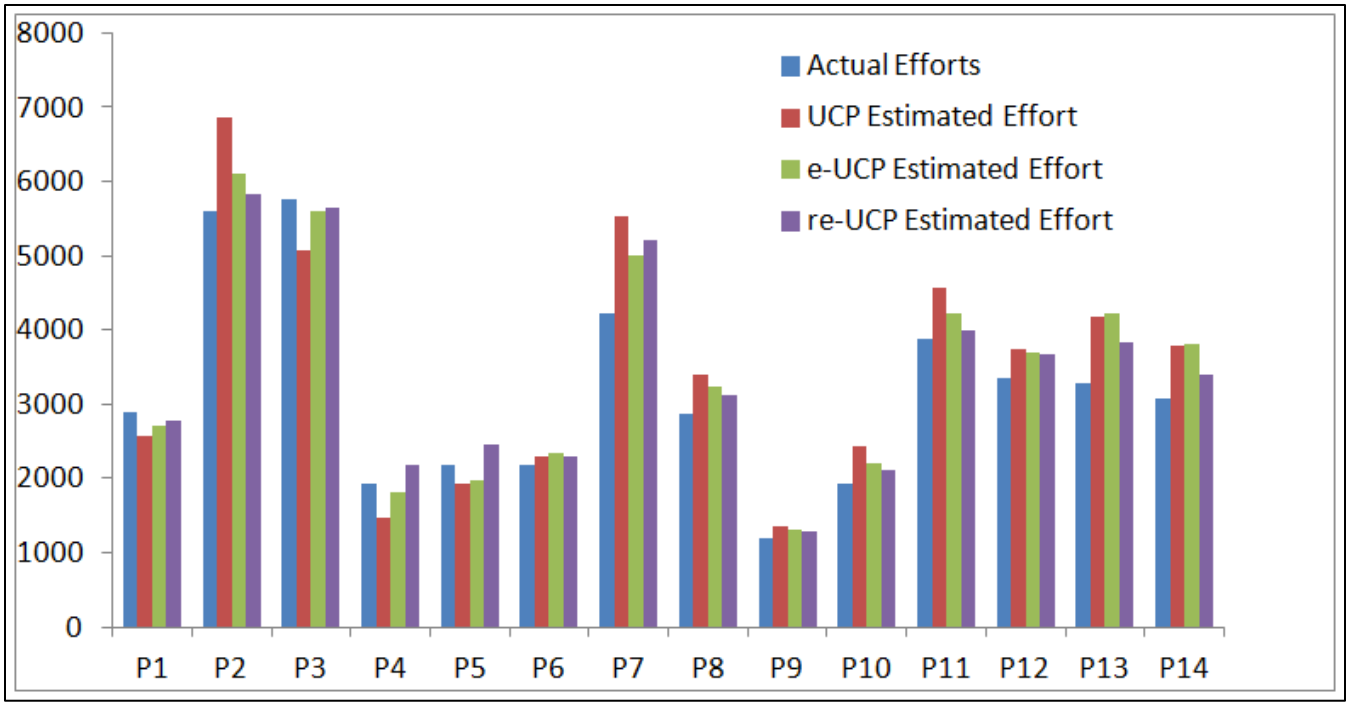


Fig. 2. Actual efforts and estimated effort using UCP, e-UCP and Re-UCP

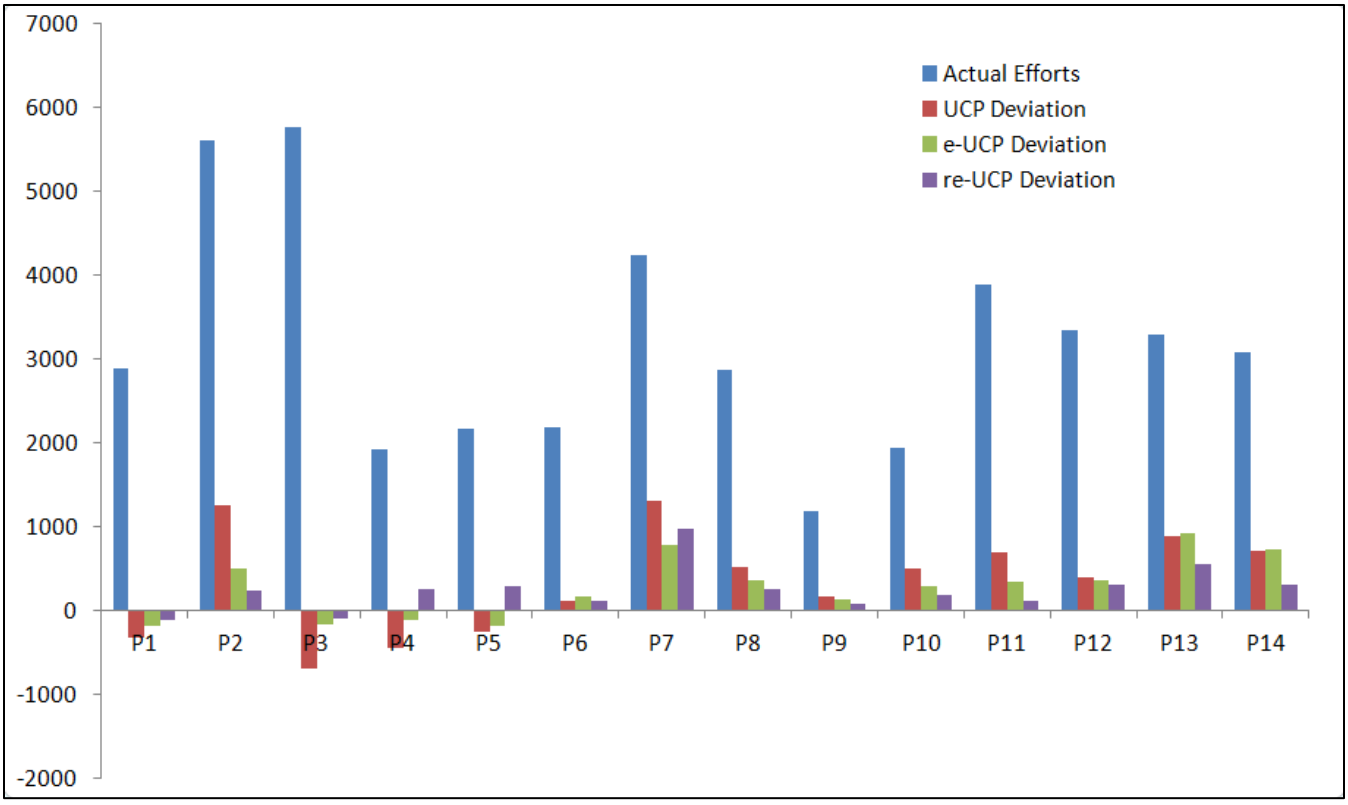


Fig. 3. Actual efforts and Deviations using UCP, e-UCP and Re-UCP

Bootstrap Approximation of Gibbs Measure for Finite-Range Potential in Image Analysis

Abdeslam EL MOUDDEN
Business and Management School
Ibn Tofaïl University
Kenitra, Morocco

***Abstract*—This paper presents a Gibbs measure approximation method through the adjustment of the associated estimated potential. We use the information criterion to prove the accuracy of this approach and the bootstrap computation method to determine the explicit form. The Gibbs sampler is the tool of our simulations while taking advantage of the use of the only one MCMC inside of the multiple necessary MCMC in the classical approximation. We focus on the validity of our approach for the Gibbs measure of a Markov Random Field with an interaction potential function and the associated uniqueness condition. Some theoretical and numerical results are given.**

Keywords—bootstrap computation; Gibbs measure; Markov Chain Monte Carlo; image Analysis; parameter estimation. Likelihood inference

I. INTRODUCTION

It is well known that in computer vision, one of the first aims is to determine, for generic prior Gibbs model, the correct form of the potential function. Certainly, this later will be better if the parametric estimation is good enough [5], but this requires both ideal conditions and expensive costs. Furthermore, the Gibbs Random Fields have become an efficient instrument in image analysis. So, the associated statistical inference has attracted a great deal of interest, because of its great adequacy in important applications related to image processing, computer vision, neural modeling and perceptual inference. Nevertheless, to estimate the true parameter for the Gibbs model required a high cost in terms of computation time and modeling conditions. For example, the use of the maximum likelihood estimation (MLE) was impossible to be calculated and was substituted by the pseudo-likelihood estimation [5] [18]. Fortunately, the intense developments in statistics accompanied by the evolution in computer systems allowed maximum likelihood estimation for Gibbs Random Fields to be constructed. For this, we propose the use of the resampling method [3] through Markov Chain Monte Carlo (MCMC). The bootstrap computation method and MCMC with Gibbs sampler is used to retrieve ever more the desired potential function for the parameter Gibbs distribution.

Some technical changes make use of one MCMC instead of two chains, which is very promising to reduce the computational time. We prove that the KL-distance is minimized for the adjusted potential function. Moreover, the adjustment method proposed in this paper keeps the features of prior potential function for the associated Gibbs measure.

This paper is organized as follows. In section 2, we present the necessary context of the Gibbs models that describes the validity of the proposed approximation. In section 3, we present the steps of the approximation proposed in this paper in particular the study of the information criterion for a Gibbs model from what we inspire the proper form of the adjusted potential and prove the accuracy of the associated approximation even though by the use of one MCMC . In section 4, we present some numerical results to explain the feasibility of the usefulness of the approximated expressions.

II. THE THEORITICAL APPROACH OF A GIBBS RANDOM FIELD WITH POTENTIAL INTERACTION FUNCTION

Before presenting the adjustment method for a parametric potential interaction function of a Gibbs distribution, it is necessary to review the concepts and results related to the Gibbs measure, which will clarify the necessity and importance of using the bootstrap approach and techniques that we introduce in this context.

A. Click and neighborhood system

It is quite obvious that a digital image is modeled by a matrix X of data on a network $S \subseteq \mathbb{Z}^2$, instead of a linear data base. The shape of such model is a Random Field $X = (X_s, s \in S)$ instead of an ARMA model for example.

In the network S , a *system of neighborhood* $\mathcal{V} = \{V_s; s \in S\}$ is defined as follows:

- $s \notin V_s$; $for\ s \in S$
- $for\ \ s,t \in S;\ s \in V_t \Leftrightarrow t \in V_s$

V_s is a set of neighboring elements of s. So, a part C of S is called a *click* with respect to $\mathcal{V}$ if:

- C is reduced to a single site
- or it contains at least two elements and each pair (s, t) of elements is formed of neighboring sites (with regard to $\mathcal{V}$).

The boundary of a non-empty subset V of the network S is a subset ∂V defined as:

$$for\ s \in \partial V\ \ there\ exists\ an\ element\ t \in V\ such\ that,\ s \in V_t$$

Given a distance d on S, the neighborhood V_s of a site s with respect to a finite-range ℓ is:

$$V_s = \{t \in S - \{t\}: d(s;t) \leq \ell\}$$

So, the boundary of a subset V of length ℓ is,

$$\partial V_\ell = \{s \in S : \exists\ t \in V\ such\ that\ d(s;t) \leq \ell\}$$

B. Random field on a nethwork

We consider a random field $X = (X_s, s \in S)$ defined on a network $S \subseteq \mathbb{Z}^2$. X takes its values in the set of configurations

$$\Omega = \Omega(S) = E^S$$

E is the set of the different levels of a pixel s. The first way to define a probability measure of the random field X on Ω is to give a Kolmogorov projective family of marginal distribution on finite subsets V of S. Nevertheless, the right way is to define a kernel family of conditional probabilities [11], which is more appropriate in case of image processing and analysis. In other words, given a probability measure ν on Ω, the conditional probability kernel on a subset V of S is defined as follows:

$$P^V(.,.) : \Omega(S-V) \times \Im(V) \longrightarrow [0,1] \quad (y\ ,\ A) \longrightarrow P^V(y,A) \qquad (1)$$

with,

$$P^V(y,A) = E_\nu(1_A/y) \qquad (2)$$

is the conditional expectation on the Borel space $\Im(V)$ given a configuration y on the outside $(S-V)$ of V, such that for all $V' \subset V$ and $A \in \Im(S)$, we have almost surely: $E_\nu[E_\nu(1_A/\Im(S-V))/\Im(S-V')] = E_\nu(1_A/\Im(S-V'))$ (3)

This can be written as the following:

$$\int P^V(x,A)P^{V'}(y,dx) = P^{V'}(y,A) \qquad (4)$$

ν −almost surely for all $y \in \Omega(S-V')$. It is expressed as an operation " $\star$ " between two kernels P and :

$$P \star Q(z,A) = \int Q(x,A)P(z,dx) \qquad (5)$$

Using the operation " $\star$ " above, we give the definition of a specification below. $fp(S)$ denotes the set of finite subsets of the network S.

Definition 1.1

A specification is a family $(\pi^V)_{V \in fp(S)}$ of probability kernels satisfying:

a) For $V \in fp(S)$ and $A \in \Im(S)$, $\pi^V(.\ ,A)$ is a random variable $\Im(S-V)$-measurable.

b) For all $V \in fp(S)$, $\pi^V(.\ ,A) \equiv 1_A$ if $A \in (S-V)$.

c) For all V and $V' \in fp(S)$, $\pi^{V'} \star \pi^V \equiv \pi^{V'}$.

Then, it follows that for a Markov Random Field given a Gibbs measure ν, the associated specification can be written as (Dobrushin-Lanford-Ruelle–equation):

for $V \in fp(S)$,

$$\pi^V(.\ ,.) = E_\nu(./\Im(S-V)) \qquad (6)$$

The sufficient condition of the existence of a Gibbs measure given a specification can be written as:

$\forall\, V \in fp(S)$ and $A \in \Im(V)$,

$$sup_{y\in\Omega} |\pi^V(y(S-V)\ ,A) - \pi^V(y(\partial_\ell V)\ ,A)| \xrightarrow[\ell\to+\infty]{} 0 \qquad (7)$$

$y(S-V)$ and $y(\partial_\ell V)$ are the restriction configurations to $(S-V)$ and $\partial_\ell V$ respectively. It should be noted that the condition (7) is true for a Markov Random Field with finite-range potential.

C. The interaction potential function of a GibbsRandom field

For a Markov Random field, the associated Gibbs measure "ν" is defined via the specification like:

$$\pi^D(\ y\ ,A) = \int_A \frac{1}{Z_D(y)} \exp[-H_D(x.y)]\ d\nu(x) \qquad (8)$$

for all $D \in fp(S)$ *and* $y \in \Omega(S-D)$. Then "ν" is a positive measure defined on Ω and $H_V(.,.)$ is the energy function given by the potential $(I_V)_{V\in fp(S)}$ such that, for $x \in \Omega(D)$ and $y \in \Omega(S-D)$:

$$H_D(x.y) = \sum_{V, V\cap D\neq\Phi} I_V(x.y) \qquad (9)$$

$(x.y)$ is the concatenation of the configuration x on D with boundary condition y on (S-D). We can recall the different results in this context; however, we are interested in conditions of existence and uniqueness of the Gibbs measure given a potential function specification. It is used in measuring the accuracy of parametric model estimation for a Markov random field. Because of this, it is introducing a quantity, for a given site s on S private the origin o:

$$\rho_s = sup\frac{1}{2}\|\pi^o(y,\ .) - \pi^s(y',\ .)\| \qquad (10)$$

The sup is taken over all configurations y and y' on $S-\{o\}$ identical everywhere except on s, and $\|.\|$ denotes the total variation norm of a measure μ, defined by:

$$\|\mu\| = sup\{|\mu(f)|;\ \|f\|_\infty = 1\}$$

The quantity ρ_s measures the maximum influence of the modality at the site s on the conditional distribution at the origin network of S. So, the uniqueness condition of a Gibbs measure [11] is of the form:

$$\sum_{s\neq o} \rho_s < 1 \qquad (11)$$

This is rewritten in [7] for a specification with a potential invariant under translation as well; $\exists\ \alpha \in [0,1[$ such that,

$$\forall\ s \in S, \quad \sum_{V\in fp(S), V\ni s}(|V|-1)\|I_V\|_\infty < \alpha \qquad (12)$$

The expression (12) is more significant and useful than (11).

D. Markov Chain Monte Carlo and paremeter estimation for Gibbs distribution

We can find some papers that dealt with the maximum likelihood estimation for a Gibbs Random Field [5] [19]. The main problem in this topic is essentially the large computation time for a complete determination of the MLE based on maximizing:

$$\pi_\theta^D(x_o/y) = \exp(-H_D(\theta, x_o.y))/z(\theta) \qquad (13)$$

for an observed configuration x_o of the Gibbs Random Field X on a subset $D \in fp(S)$, where y is a boundary configuration outside D. the expression (13) is the parametric version of (8) with respect to the parametric potential $(I_V(\theta,.))_{V\in fp(S)}$. The normalization constant is:

$$z(\theta) = \sum_{x\in\Omega(D)} \exp(-H_D(\theta, x.y))$$

Recently the construction of the MLE is allowed due to computational system evolution. It is to solve the derivative equation:

$$E_\theta(\nabla H_D(\theta, X) - \nabla H_D(\theta, x_o) = 0 \qquad (14)$$

∇ represents the gradient operator of partial derivative with respect to the parameter coordinates $\theta \in \Theta \subseteq \mathbb{R}^p$ where $p \geq 1$. The solution $\widehat{\theta}$ (i.e. the MLE) of (14) in θ is approximated by the limit of a stochastic sequence through a dynamic markovian algorithm using a MCMC $(X^{j,n})_{1\leq j\leq N}$ via a Gibbs sampler controlled by the θ_n term of the sequence:

$$\theta_{n+1} = \theta_n + \alpha\,(\hat{E}_{\theta_n}(\nabla H_D(\theta_n, X) - \nabla H_D(\theta_n, x_o)) \qquad (15)$$

where,

$$\hat{E}_{\theta_n}(\nabla H_D(\theta_n, X) = \sum_{j=1}^{N} \nabla H_D(\theta_n, X^{j,n})\,/N$$

And it is allowed under the Gibbs sampler ergodic property.

III. Adjustment of the Interaction Potentiel Functiion of a Gibbs Measure

In this section, we to introduce the bootstrap methodology [4], as an adjustment tool for the parametric energy function $H_D(\hat{\theta}, x.y)$, given in (13), to obtain a more appropriate Gibbs measure compared to that associated with the estimated potential. This is proved using the information criterion.

A. Information Criterion and Variational Principle of Gibbs Random fields

In the framework of the modeling of a random field on a regular network, it is proved that the variational principle [21], i.e. the decision to assign a Gibbs measure "ν" of a random field X when the true one is "μ", can be expressed in terms of an information criterion $h(\nu, \mu)$, and this vanishes if and only if $\mu = \nu$. The same result is given [8] in the restricted case; the information criterion $h(\nu, \mu)$ vanishes if and only if μ and ν have the same interaction potential.

For this reason it is obvious that the true Gibbs measure is not necessarily to be associated with the estimated potential $(I_V(\hat{\theta}, .))_{V\in fp(S)}$; the fact that the maximum likelihood estimator for the Gibbs fields is substantially biased [19], which is the same case for the pseudo likelihood estimator [5]. So, the following adjustment of the estimated potential proposed in this paper has a considerable magnitude to dig up the accurate specification of the parametric Gibbs measure.

Doing this, we consider firstly the kullback-Leibler function applied to the parametric Gibbs measure P_θ and to another any Gibbs measure Q,

$$K(P_\theta, Q) = \lim_{D} \inf K_D(P_\theta, Q)$$

Where,

$$K_D(P_\theta, Q) = \begin{cases} E_\theta\left[\log\left[\frac{dP_{\theta,D}}{dQ_D}\right]\right], & if\ P_{\theta,D} \ll Q_D \\ \infty, & otherwis \end{cases} \qquad (16)$$

The symbol " $\ll$ " means that P_θ is absolutely continuous with respect to Q_D. $\frac{dP_{\theta,D}}{dQ_D}$ is the Radon–Nikodym derivative of the two restricted measures on the finite subset D of the lattice S. The specific information of P_θ with respect to Q is:

$$h(P_\theta, Q) = \lim_D \inf \frac{1}{|D|} K_D(P_\theta, Q) \qquad (17)$$

We recall that if P_θ is the Gibbs measure of the specification $(\pi^V)_{V\in fp(S)}$ then, $h(P_\theta, Q) = 0$ if and only if Q has the same specification $(\pi^V)_{V\in fp(S)}$. It means that for a specification with a finite-range interaction potential, $h(P_\theta, Q) = 0$ if and only if P_θ and Q have the same interaction potential.

B. Adjustement of the Gibbs measure potential

For a parametric Gibbs model, we note the following condition:

$$\theta_1 \neq \theta_2 \Longrightarrow G(\theta_1) \cap G(\theta_2) = \Phi \qquad (18)$$

It means that two different values of θ give two different Gibbs measures. So, under this condition [18] we have the following result:

$$\mathrm{Lim}_{D\nearrow S} \frac{1}{|D|} log\left[\frac{\pi^D_{\theta_r}(x)}{\pi^D_{\theta}(x)}\right] = -h(P_{\theta_r}, P_\theta) = -h(\theta_r, \theta) \qquad (19)$$

which exists almost surely, where θ_r is the true value of the parametric Gibbs model for a Markov Random Field X observed through an image x on the finite subset D. So,

$$h(\theta_r, \theta) = 0 \iff \theta = \theta_r \qquad (20)$$

Furthermore, if Θ is a compact space of $\mathbb{R}^p$, there exists a constant $\beta \geq 0$, such that, for all θ_1 and θ_2 in Θ:

$$\frac{1}{|D|}\left|log\left[\pi^D_{\theta_1}(x)\right] - log\left[\pi^D_{\theta_2}(x)\right]\right| \leq \beta.|\theta_1 - \theta_2| \qquad (21)$$

In practice, the observation window D is quite low with respect to the lattice S, in addition, the MLE has significant bias. So, the estimated value $\theta_1 = \hat{\theta}$ is not so satisfying, but from (19) we can note for an estimation θ_1 of θ_r:

$$log\left[\frac{\pi^D_{\theta_r}(x)}{\pi^D_{\theta_1}(x)}\right] \cong -|D|.h_D(\theta_r, \theta_1) \qquad (22)$$

Also, from (9) and (13) we have:

$$\pi^D_{\theta_r}(x) \cong \frac{exp[-\sum_{V\cap D\neq\Phi} I_V(\theta_1, x)]}{Z_D(\theta_1)} \times exp[-|D|.h_D(\theta_r, \theta_1)] \qquad (23)$$

Then, this logical approximation induces the adjusted potential given an estimation θ_1 of the unknown true value of the parameter Gibbs model as follows:

$$\tilde{I}_V(x) = \begin{cases} I_V(\theta_1, x) + \frac{|D|}{|\phi_D|}.h_D(\theta_r, \theta_1), & if\ diam(V) \leq \gamma \\ 0, & otherwise \end{cases} \qquad (24)$$

where γ denote the finite-range of the potential $(I_V)_V$ and $\phi_D = \{V \in fp(S): V\cap D \neq \Phi\ and\ diam(V) \leq \gamma\}$ .So, it is easy to verify that $(\tilde{I}_V)_V$ is an interaction potential. This potential verifies the condition of Dobrushin's theorem for the uniqueness of Gibbs measure (12) as well as the estimated potential $(\mathrm{I_V}(\theta_1, .))_\mathrm{V}$. Indeed, from (21) and (24) we can write the following inequality for all $V \in fp(S)$:

$$\left\|\tilde{I}_V\right\|_\infty \leq \|I_V(\theta_1, .)\|_\infty + \frac{|D|}{|\phi_D|}\beta.\|\theta_r - \theta_1\|_p, \qquad (25)$$

On the other hand, for a site s $\in S$, we have:

$$\sum_{V\ni s}(|V|-1)\left\|\tilde{I}_V\right\|_\infty \leq \alpha + \beta.\sum_{V\ni s}(|V|-1)\frac{|D|}{|\phi_D|}\|\theta_r - \theta_1\|_p$$

Also,

$$\sum_{V \ni s} (|V|-1)\left\|\tilde{I}_V\right\|_\infty \leq \alpha + \beta.(\gamma^2-1)\frac{|D|}{|\phi_D|}|\phi_{\{s\}}|.\|\theta_r-\theta_1\|_p$$

It follows that the adjusted potential $(\check{\mathrm{I}}_V)_V$ satisfies uniqueness condition if:

$$\|\theta_r-\theta_1\|_p \leq (1-\alpha)\frac{|\phi_D|}{|D|.|\phi_{\{s\}}|}[\beta.(\gamma^2-1)]^{-1} = \varepsilon$$

Note that for $\gamma = 1$, the above result is trivial; otherwise, from the consistency of the Maximum Likelihood Estimator we can obtain for a sufficiently large window D, $\|\theta_r-\theta_1\|_p \leq \varepsilon$.

C. The Approximated Gibbs Measure

We note by $\tilde{P}$ the Gibbs measure associated to the adjusted potential $(\check{\mathrm{I}}_V)_V$. The initial parametric Markov random field X has almost surly the same previous potential. In fact, the specification information in (17) is minimized by construction as in (22). Then, if we note $(\tilde{\pi}^D)_D$ the specification associated to $(\check{I}_V)_V$ in the sense of (8) and (9), we have exactly:

$$\liminf_D h_D(P_{\theta_r}, \tilde{P}) = 0$$

What implies that, under the uniqueness condition, the two Gibbs measures P_{θ_r} and $\tilde{P}$ are the same. In the case of the non-uniqueness, P_{θ_r} and $\tilde{P}$ have at least the same interaction potential function.

D. Bootstrap Approximation of Gibbs measure

It is clear that the approximation Gibbs measure depends on the unknown value θ_r of the parameter Gibbs model. So, we propose the use of the Bootstrap methodology [3], which is successfully developed in different areas of applied statistic. It is to estimate the quantity $h_D(\theta_r, \theta_1)$ in (24) by $h_D(\theta_1, \theta_1^*)$, which is completely determined given the same initial observed realization x of the Markov Random Field X. The value θ_1^* is the bootstrap estimation of the unknown value θ_r given the same initial observed configuration x of X. because of this, we introduce the Bootstrap approximation of the adjusted interaction potential $(\check{\mathrm{I}}_V)_V$ as follow:

$$\tilde{I^*}_V(x) = \begin{cases} I_V(\theta_1, x) + \frac{|D|}{|\phi_D|}.h_D(\theta_1, \theta_1^*), & if\ diam(V) \leq \gamma \\ 0\ , & otherwise \end{cases} \quad (26)$$

And we use the ergodic approximation of the MCMC $(X^j)_j$ with invariant distribution μ for the expected quantity as:

$$\hat{E}_\mu(g) = \frac{1}{N}\sum_{j=1}^{N} X^j$$

However, to obtain an approximation of the quantity $h_D(\theta_1, \theta_1^*)$ in (26), we need more than two MCMC, due to the fact that we have as in (16) and (17):

$$|D|.h_D(\theta_1, \theta_1^*) = E_{\theta_1}\left[\log\left(\frac{\pi^D_{\theta_1}(X)}{\pi^D_{\theta^*_1}(X)}\right)\right] \quad (27)$$

what requires a important computational time. Fortunately, we can write easily the following expression:

$$\frac{z_D(\theta_1)}{z_D(\theta_1^*)} = E_{\theta_1^*}[exp(\Delta H_D)] \quad (28)$$

Where,

$$\Delta H_D = \Delta H_D(X) = H_D(\theta_1^*, X) - H_D(\theta_1, X)$$

Also,

$$E_{\theta_1}\left[\log\left(\frac{\pi^D_{\theta_1}(X)}{\pi^D_{\theta^*_1}(X)}\right)\right] = E_{\theta_1^*}\left[\frac{\pi^D_{\theta_1}(X)}{\pi^D_{\theta^*_1}(X)}.\log\left(\frac{\pi^D_{\theta_1}(X)}{\pi^D_{\theta^*_1}(X)}\right)\right] \quad (29)$$

The right value in (27) may be finally written as the following expression:

$$E_{\theta_1}\left[\log\left(\frac{\pi^D_{\theta_1}(X)}{\pi^D_{\theta^*_1}(X)}\right)\right] = \frac{E_{\theta_1^*}\{(\Delta H_D).exp[\Delta H_D]\}}{E_{\theta_1^*}[exp(\Delta H_D)]} - log\{E_{\theta_1^*}[exp(\Delta H_D)]\}$$

Thus, we get the following approximation of the adjustment term in (24):

$$|D|\tilde{h}_D(\theta_1, \theta_1^*) = \sum_{n=1}^{N} c(n).\Delta H(X^n) - log\left[\frac{1}{N}\sum_{n=1}^{N}\exp\left(\Delta H(X^n)\right)\right] \quad (30)$$

The $c(n)$ coefficient is determined by:

$$c(n) = \frac{\exp(\Delta H(X^n))}{\sum_{j=1}^{N}\exp(\Delta H(X^j))} \quad (31)$$

Using an unique MCMC $(X^j)_{j=1,\ldots,N}$ simulated by the Gibbs sampler under the invariant measure $\pi_{\theta_1^*}$.

IV. Simulation Examples

To implement our approximation we consider an interaction potential function for a parametric Gibbs model with a finite-rang $\gamma = 2$ and has an invariant interaction coefficient in a click with regard to a system of neighborhood $\mathcal{V}$. So, the model parameter θ is in $\Theta \subset \mathbb{R}^1$. Firstly we expose the simulated MCMC for this mode in different stats, followed by a numerical computation of the adjusted term. Finally we examine graphically the main property of the approximated quantity $\tilde{h}_D(\theta_1, \theta_2)$.

a) MCMC silmulation example :

From this Gibbs model and for a numeric value of θ, we put up different configurations generated by the Gibbs sampler. We obtain the different stats at the given moments (Fig.1.).

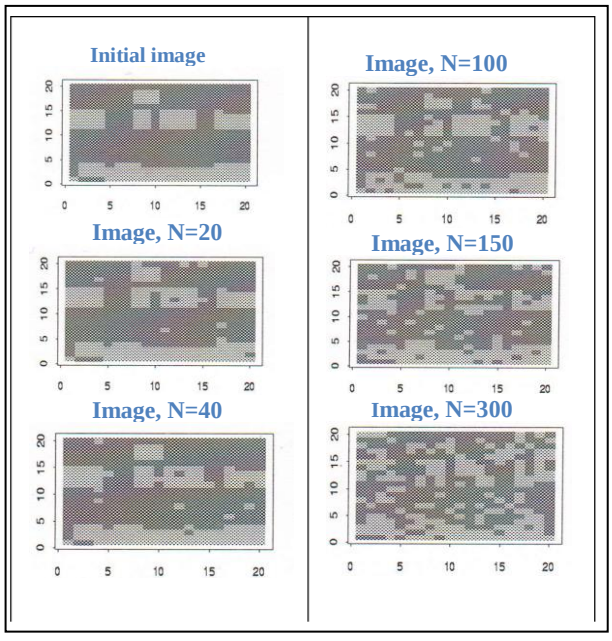


Fig. 1. Simulated stats of MCMC for a finite-range interaction potential

b) Example of the approximation function $\tilde{h}_D(\theta_1, \theta_1^*)$:

For the same Gibbs model above, we calculate the approximation function $\tilde{h}_D(\theta_1, \theta_2)$ for two various values θ_1 et θ_2, to examine the evolution of this function with regard to the difference between θ_1 et θ_2, using the MCMC simulated until different two states(N=600 and N=1000). The case indicated by NA means that the denominator of $c(n)$ in (31) becomes a very high number.

TABLE I. THE VALUE OF $\tilde{h}_D(\theta_1, \theta_2)$ FOR DIFFERENT VALUES OF θ_1 ET θ_2

Value of θ_1	Value of θ_2	$\lvert\theta_1 - \theta_2\rvert$	N=600	N=1000
-10	-9	1	0.0139	0.0116
-2	-1.5	0.5	0.0077	0.0102
-1	-0.23	0.77	0.0112	0.0086
-0.1	-0.002	0.098	0.0039	0.0038
-0.3	0	0.3	0.0054	0.008
0	0.5	0.5	0.0104	0.0106
0.1	0.9	0.8	0.0166	0.0098
0.3	1	0.7	0.0117	0.0110
1	2	1	0.0162	0.0133
2	3	1	0.0125	0.0106
10	11	1	0.0120	0.0154
10	12	2	0.01704	0.01593
10	15	5	NA	NA

This gives the shape of the two graphs of this function as follows:

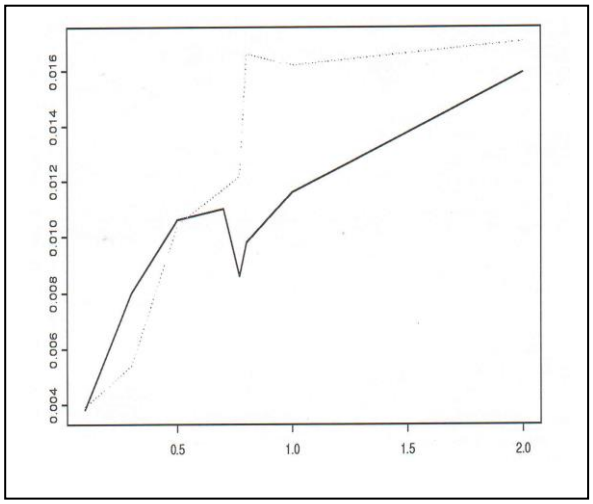


Fig. 2. The variation of the approximated function based on the difference between $\theta_1$and θ_2; the dotted-line curve for N=600 stats of the MCMCm and the continuous-line curve for N=1000

We can notice that the approximate function keeps the theoretical properties of the initial function, such as

$$\tilde{h}_D(\theta_1, \theta_2) \cong 0 \text{ if } \theta_1 \cong \theta_2$$

The main advantage of this approximation is in its easy use while keeping its strong properties as information criterion and adjusted term for the estimated potential. This assures the validity of the approximation of the Gibbs measure given above.

REFERENCES

[1] A. Benveniste, P. Metivier, Priouret, "Algorithmes adaptatifs et approximations stochastiques, théorie et application-techniques stochastiques". Masson, 1987.

[2] B.D. Ripley, "Spacial statistics". Wiley, New York. 1981

[3] B. Efron, "Bootstrap methods: Another look at Jackknife", Annals of Statistic, 7, 1-26. 1979.

[4] B. Efron, and Tibshirani, R., "An introduction to the Bootstrap". Monographs on statistics and applied Probability 57, Chapman and Hall. 1993.

[5] B. Gidds, "Parameter estimation for Gibbs distributions from fully observed data. In Markov Ranfom Fields: theory and applications". Cellapa, L. and Jain, A. (eds). 741-498. Academic Press, New York. 1993.

[6] B. Prum, "Processus sur un réseau et mesure de Gibbs; application". Techniques stochastiques, Masson. 1986.

[7] B. Simon, " A remarque on Dobrushin uniqueness theorem". Comm. Math-Phys. p.183. 1979

[8] C. Preston, "Random fields". In lect.nots in Math, vol. 534. Berlin, hedlber, New York, Springer. 1976.

[9] D Geman, and S. Geman, "Stochastic Relaxation, Gibbs Distribution and Baysian Restoration of Images". IEEE TPAM, vol-6 pp721-741. 1984.

[10] S. Singh, "On the asymptotic accuracy of Efron's Bootstrap". The Annals of Statistics vol. 9, N°6, 1187-1195. 1981.

[11] L. Dobrushin, "The description of a random field by means of conditional probabilities and it conditions of its regularity". Theory. Prob. Appl. XIII-2 pp 197-224. 1968.

[12] L. Tierney, " Markov Chains for exploring posterior distributions". Tech. Rep. 560, School of statistics, University of Minnesota. 1991.

[13] J. Besag, "Spatial interaction and the statistical analysis of lattice systems". J. Roy. Soc. Ser B, 36 192-236. 1974.

[14] P. Hall, "On the Bootstrap and Edgeworth expansion". New York, Spring. 1992.

[15] X. Guyon, "Champs aléatoires sur un réseau". Techniques Stochastiques, Masson. 1993.

[16] V. Ripley, "Modern Applied Statistics with Splus". Springer, New York, Berlin. 1996.

[17] W. K. Hastings, "Monte Carlo sampling methods using Markov Chain and their application". Biometrika, 57, 97-109. 1970.

[18] L. Younes, "Parametric inference for imperfectly observed Gibbsian fields. Prob. Theory related fields". 82 pp 625-645. 1989.

[19] L. Younes, "Maximum Likelihood Estimation for Gibbsian Fields". Lec. Notes-Monograph series, vol. 20. Spatial statistics and Imaging, pp 403-426. 1991

[20] F. Comets, B. Gidas, "Parameter Estimation for Gibbs Distributions from Partially observed Data". The Annals of App. Probability, vol. 2, pp 142-170. 1992.

[21] H. Föllmer, J.L .Snell, " An Inner Variational Principle for Markov Fields on Graph", Gebiet, 39 pp 187-195. 1977.

Jigsopu: Square Jigsaw Puzzle Solver with Pieces of Unknown Orientation

Abdullah M. Moussa
Electrical Engineering Department
Faculty of Engineering
Port-Said University, Port-Said, Egypt

***Abstract*—In this paper, we consider the square jigsaw puzzle problem in which one is required to reassemble the complete image from a number of unordered square puzzle pieces. Here we focus on the special case where both location and orientation of each piece are unknown. We propose a new automatic solver for such problem without assuming prior knowledge about the original image or its dimensions. We use an accelerated edge matching based greedy method with combined compatibility measures to provide fast performance while maintaining robust results. Complexity analysis and experimental results reveal that the new solver is fast and efficient.**

Keywords—jigsaw puzzle; image merging; edge matching; jigsaw puzzle assembly; automatic solver

I. Introduction

It is generally accepted that the first modern jigsaw puzzle was built in 1760 by London map maker John Spilsbury for educational purposes [1]. Since then, several different manufacturers around the world are manufacturing jigsaw puzzles in many shapes, sizes and piece types. The jigsaw puzzle is provably technically challenging. It has been shown by Demaine et al. [2] that the jigsaw puzzle problem is NP-complete in the general case when the pairwise affinity of jigsaw pieces is unreliable. The computational problem of jigsaw puzzle assembly was first introduced nearly fifty years ago in a fundamental work by Freeman and Gardner [3]. In addition to being an interesting problem in its own right, the computational jigsaw assembly has many applications in recovering shredded documents or photographs [4, 5, 6, 7], reassembling archaeological artifacts [8], DNA/RNA modeling [9] and speech descrambling [10].

Many attempts have been made to handle the problem. Several papers [11, 12] assume using classic jigsaw pieces with distinct shapes, and focus on matching the shape of the pieces to solve the puzzle. Some others use both of image contents and boundary shape [13, 14, 15]. In this paper, we follow the lead of recent work [16, 17, 18] and consider jigsaw puzzles with square pieces. We believe that assuming prior knowledge about the dimensions of the complete image of the puzzle pieces, as what the majority of the existing algorithms do, can reduce the applicability of the algorithm used. So, we relax the condition that the dimensions of the complete image should be previously known. We focus on the special case where both location and orientation of each piece are unknown. We present a new fast algorithm to tackle such problem. Our algorithm, named JigSoPU (Jigsaw Solver with Pieces of Unknown orientation), uses an edge matching based greedy technique along with a combined compatibility score functions to provide an accelerated performance while maintaining robust results. Complexity analysis and experimental results show that the new algorithm is fast and efficient.

The rest of the paper is organized as follows: Section II introduces the new jigsaw assembly algorithm. Section III presents the complexity analysis of the proposed algorithm. In Section IV the experimental results are presented. Finally, conclusions are summarized in Section V.

II. Algorithm Description

To solve the square jigsaw puzzle problem, we need to consider two aspects, a criterion of compatibility between jigsaw pair of pieces, and a specific strategy to assemble the pieces. In section II-A, we will present the compatibility measure used, and in section II-B the new edge matching based assembly strategy will be proposed.

A. Pairwise Compatibility Criterion

When the jigsaw puzzle is correctly assembled, it can be observed that the adjoining pieces have often adjacent edges with pixels of similar intensity values. Such feature is the base of jigsaw edge matching based solvers. Common dissimilarity measures can be used to calculate the minimum difference between the pixels of pieces' edges in search of the best match between two candidate pieces. However, depending on a single dissimilarity measure may make the algorithm getting trapped in an incorrect assembly. So, we propose to use a combined measure that jointly minimizes the mean value of SAD (sum of absolute difference) of candidate adjacent edges along with the amount of pixel pairs that have a SAD value above a predefined threshold. We found experimentally that using the proposed combined compatibility functions is efficient enough to even handle color images after converting them to grayscale versions without a need to deal with each color channel alone.

B. Assembly Strategy

In this section, we introduce the new edge matching based greedy assembly algorithm. The algorithm is inspired by the FRoTeMa algorithm [19, 20] for template matching. Exhaustive matching of edges in search of the complete image should be robust, but it is not efficient in time because such procedure will have a complexity of $O(N^2)$ where N is the number of edges. This costly complexity makes the exhaustive matching procedure difficult to be applied to large jigsaw

puzzles. So, there is always a need for faster algorithms that can tackle the problem without sacrificing robustness.

In order to be able to provide a faster solver, we should reduce the search space of the problem. We did this based on the observation mentioned in the previous section that the adjoining pieces in a correctly solved jigsaw puzzle have often similar intensity values of the pixels in the adjacent edges. Based on this observation, the sums of pixel intensities of adjacent edges tend to be similar. The following steps are made to make use of the last concept:

1) Calculate sum of pixel intensities for each edge in all pieces. We will call such sum as the weight of the edge.

2) Store the calculated weights in a vector S_v and then sort the vector in ascending order.

3) Pick the edge associated with a random weight in S_v.

4) Assign the picked edge to e_{ct}, its weight index in S_v to i_{ct} and the piece associated with it to p_{ct}.

Using a predefined threshold t_c, the sorted vector S_v can be used to check the possible match between e_{ct} and a subset of the edges within the range of indices in S_v of: $i_{ct} \pm t_c$. Within the mentioned range, each edge is checked for a possible match with e_{ct} using the criterion described in section II.A, if one of the edges meets the requirements of the compatibility measure, the piece associated with such edge will be a *candidate piece* p_{cand}. To even increase the robustness of the process, not all candidate pieces are considered as correct pieces, Referring to Fig. 1, the piece in dark blue is an example of p_{ct} and the piece in light blue is an example of p_{cand}.

To ensure that p_{cand} is a correct match with p_{ct}, we check the compatibility of the pieces around p_{ct} and p_{cand} from the two sides that are perpendicular to e_{ct} (the pieces in gray in Fig. 1) using the same strategy. If this patch of pieces is compatible, i.e., each piece of them and its candidate adjoining ones meet the requirements of the compatibility measure, in this case p_{cand} will be considered as a correct match and will be assigned to p_{ct}. If such patch of pieces is not compatible or none of the edges in the range $i_{ct} \pm t_c$ meets the requirements of the compatibility measure, the algorithm will consider p_{ct} as a *boundary piece*, i.e., p_{ct} is considered to have no adjacent pieces from the side of e_{ct}.

In this case and based on the direction of the blue arrow, one of the two perpendicular edges to e_{ct} will be the new e_{ct}. The process is continued sequentially using the direction of the blue arrow until assembling the rest of the pieces or reaching the corner, i.e., when reaching a piece which is a boundary piece from two sides. If the algorithm reaches a corner, the process is continued sequentially from the other side of the first picked piece (the pink arrow) until assembling the rest of the pieces.

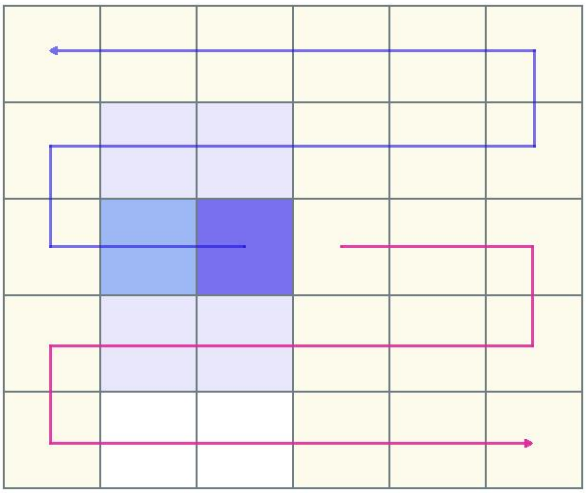


Fig. 1. Overview of JigSoPU assemply strategy

III. Complexity Analysis

Let N be the number of edges of all pieces, M the number of pixels in each edge and C the number of candidate adjacent edges checked for each edge. The operations used to calculate S_v have a complexity of $O(NM)$, while the operations used to sort S_v have a complexity of $O(N\ log\ N)$. Also, the algorithm uses $O(NMC)$ computations to assemble pieces. Thus, the overall complexity of the algorithm is $O(N\ (M\ C + log\ N))$. Such complexity makes a big difference with respect to speed of computation in comparison with the quadratic complexity we get when we match edges exhaustively.

IV. Experimental Evaluation

To check the performance of the new algorithm, we have applied the algorithm on 10 images from ETHZ dataset [21]. All experiments have been run on a Core i-5 (2.3-GHz PC) with 4 GB of RAM.

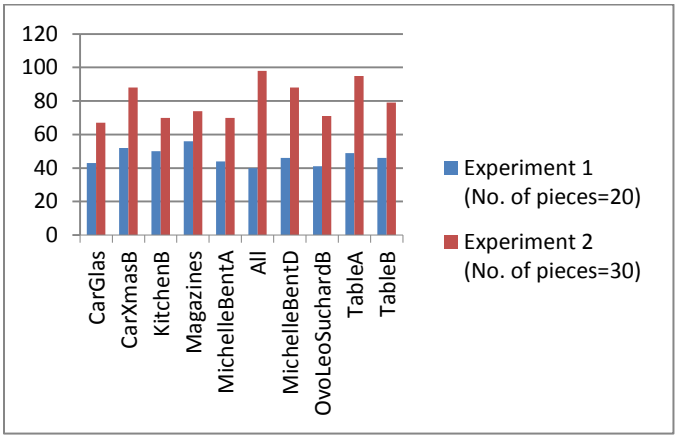


Fig. 2. Execution time of JigSoPU in milliseconds for each experiment

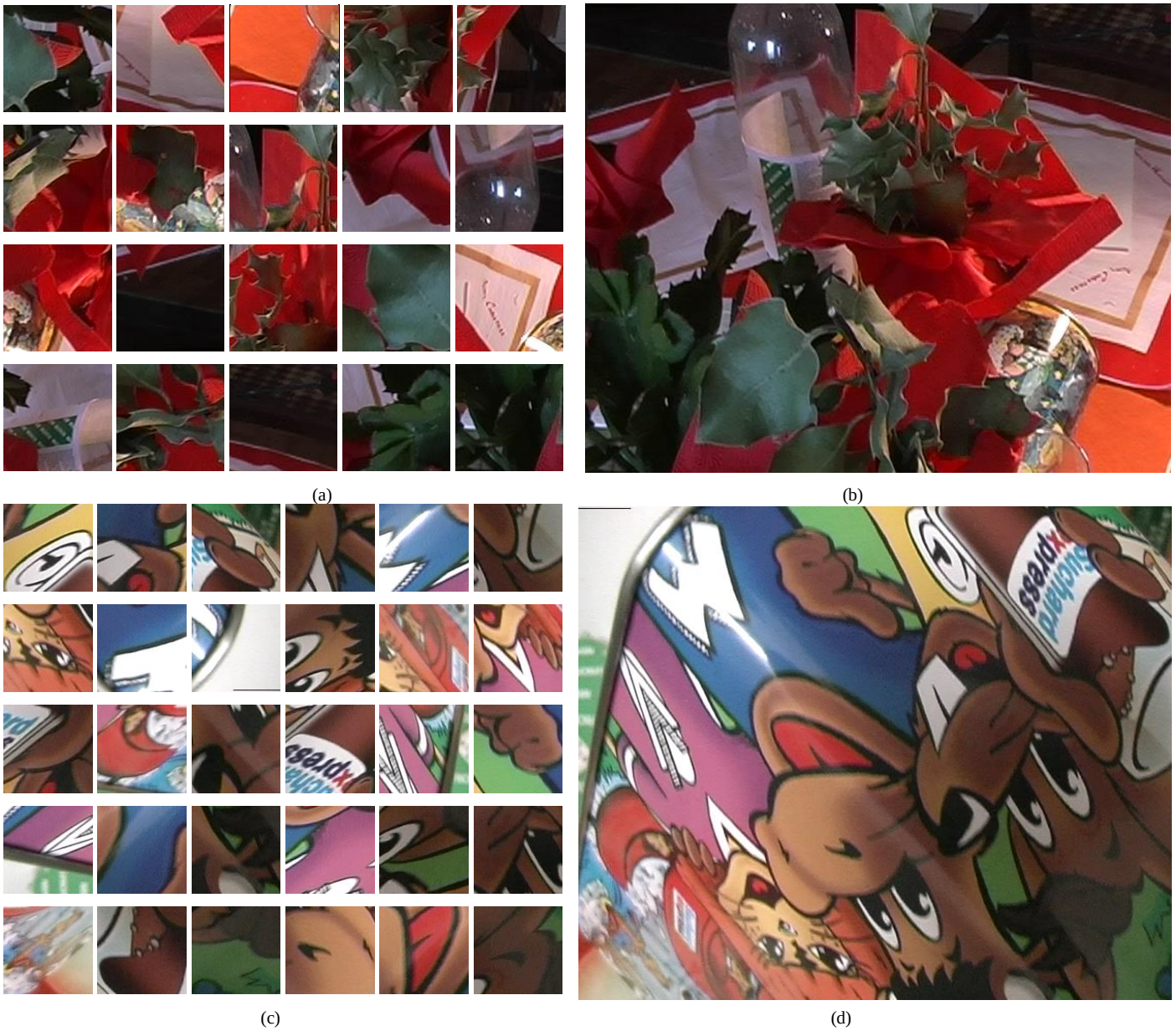


Fig. 3. Some examples of JigSoPU output. In (a) and (c), jigsaw puzzles of 20 and 30 pieces respectively. Perfect assembly is achieved in (b) and (d)

The algorithm has been tested twice for each image, in the first experiment, each image is split into 20 unordered randomly oriented square pieces, while in the second experiment, each image is split into 30 pieces in a similar manner. The proposed algorithm exhibited 100% accuracy for all of the tested images.

We have calculated the execution time for each experiment. Fig. 2 summarizes the performance of the proposed algorithm for each tested image. As shown in Fig. 2, the proposed algorithm provides fast computation as we can expect regarding its computational complexity. Some examples of JigSoPU performance are illustrated in Fig. 3. As we can see in Fig. 3, JigSoPU can assemble the complete images perfectly.

V. CONCLUSION

Computational jigsaw assembly is an interesting problem, which plays a vital role in many different scientific fields, such as image processing, computer vision, archeology and genomics. In this paper, we have presented a new fast algorithm to handle the problem of assembling square jigsaw puzzles where both location and orientation of each piece are unknown and without assuming prior knowledge about the complete image or its dimensions. The proposed algorithm has been tested against different images using different number of pieces. Complexity analysis and experimental results reveal that the new algorithm can provide fast computation and robust results.

Future work includes verifying the performance of the new algorithm against larger images and checking the effect of increasing the number of puzzle pieces on the performance. Also, it will be interesting to evaluate the applicability of the method to archaeological and genomic datasets.

REFERENCES

[1] R. Tybon. Generating Solutions to the Jigsaw Puzzle Problem. PhD thesis, Griffith University, Australia, 2004.

[2] E. D. Demaine and M. L. Demaine. "Jigsaw puzzles, edge matching, and polyomino packing: Connections and complexity," Graphs and Combinatorics, 23, 2007.

[3] H. Freeman and L. Gardner. "Apictorial jigsaw puzzles: The computer solution of a problem in pattern recognition," IEEE. Trans. on Electronic Computers,1964.

[4] S. Cao, H. Liu, and S. Yan. "Automated assembly of shredded pieces from multiple photos," In Proc. ICME, 2010.

[5] E. Justino, L. Oliveria, and C. Freitas. "Reconstructing shredded documents through feature matching," Forensic Science International, 2006.

[6] M. Marques and C. Freitas. "Reconstructing strip-shredded documents using color as feature matching," Proc. ACM Symposium on Applied Computing, 2009.

[7] L. Zhu, Z. Zhou, and D. Hu. "Globally consistent reconstruction of ripped-up documents," IEEE TPAMI, 2008.

[8] B. J. Brown, C. Toler-Franklin, D. Nehab, M. Burns, D. Dobkin, A. Vlachopoulos, C. Doumas, and T. W. Szymon Rusinkiewicz. "A system for high-volume acquisition and matching of fresco fragments: Reassembling theran wall paintings," ACM TOG (SIGGRAPH), 2008.

[9] W. Marande and G. Burger. "Mitochondrial DNA as a genomic jigsaw puzzle," Science, 2007.

[10] Y.-X. Zhao, M.-C. Su, Z.-L. Chou, and J. Lee. "A puzzle solver and its application in speech descrambling," In ICCEA,2007.

[11] D. Goldberg, C. Malon, and M. Bern. "A global approach to automatic solution of jigsaw puzzles," In Symposium on Computational Geometry, 2002.

[12] W. Kong and B. B. Kimia. "On solving 2D and 3D puzzles using curve matching," In IEEE CVPR, 2001.

[13] F.-H. Yao and G.-F. Shao. "A shape and image merging technique to solve jigsaw puzzles," PRL, 2003.

[14] T. R. Nielsen, P. Drewsen, and K. Hansen. "Solving jigsaw puzzles using image features," PRL, 2008.

[15] M. Makridis and N. Papamarkos. "A new technique for solving a jigsaw puzzle," In IEEE ICIP, 2006.

[16] N. Alajlan. "Solving square jigsaw puzzles using dynamic programming and the hungarian procedure," Amer. Journ. Applied Sciences, 2009.

[17] X. Yang, N. Adluru, and L. Latecki. "Particle filter with state permutations for solving image jigsaw puzzles," In Proc. CVPR, 2011

[18] A.C., Gallagher, "Jigsaw puzzles with pieces of unknown orientation," in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Comp. Soc. Press, Los Alamitos, CA, 2012, pp. 382–389.

[19] A. M. Moussa, M.I. Habib, and R. Y. Rizk. "FRoTeMa: Fast and Robust Template Matching," , unpublished.

[20] A. M. Moussa, Robust Template Matching with Rotation, Scale, Brightness and Contrast Invariance. Master's thesis, Port-Said University, Egypt, 2015.

[21] V. Ferrari, T. Tuytelaars, and L. V. Gool, "Simultaneous object recognition and segmentation from single or multiple model views," Int. J. Computer Vision, 2006.

Construction of FuzzyFind Dictionary using Golay Coding Transformation for Searching Applications

Kamran Kowsari[†], Maryam Yammahi[†], Nima Bari[†], Roman Vichr[*], Faisal Alsaby[†], Simon Y. Berkovich[†]

[†] Department of Computer Science,
School of Engineering and Applied Sciences at
The George Washington University,
Washington DC, 20052

[*] Exprentis, Inc
Department of Data Mining and Engineering
Fairfax VA, 22030

***Abstract*—searching through a large volume of data is very critical for companies, scientists, and searching engines applications due to time complexity and memory complexity. In this paper, a new technique of generating FuzzyFind Dictionary for text mining was introduced. We simply mapped the 23 bits of the English alphabet into a FuzzyFind Dictionary or more than 23 bits by using more FuzzyFind Dictionary, and reflecting the presence or absence of particular letters. This representation preserves closeness of word distortions in terms of closeness of the created binary vectors within Hamming distance of 2 deviations. This paper talks about the Golay Coding Transformation Hash Table and how it can be used on a FuzzyFind Dictionary as a new technology for using in searching through big data. This method is introduced by linear time complexity for generating the dictionary and constant time complexity to access the data and update by new data sets, also updating for new data sets is linear time depends on new data points. This technique is based on searching only for letters of English that each segment has 23 bits, and also we have more than 23-bit and also it could work with more segments as reference table.**

Keywords*—*FuzzyFind Dictionary; Golay Code; Golay Code Transformation Hash Table; Unsupervised learning; Fuzzy search engine; Big Data; Approximate search; Informational Retrieval; Pigeonhole Principle; Learning Algorithms ; Data Structure

I. INTRODUCTION

The interaction between a search engine and database requires that the database structure to be consistent with a search engine to search quickly, easily, and efficiently. Golay Coding clustering technique which has faster time complexity in comparison with previous conventional methods such as K-means, spectral clustering and hierarchical clustering [8, 11] . Also the traditional method of clustering cannot cover fuzzy logic. This method is used for error correction, clustering, and other aspect in computer science. Our aim is to first modify how we utilize the Golay Code Clustering Hash Table (GCCHT), generates the Golay Code Transformation Hash Table (GCTHT), and finally creating the FuzzyFind Dictionary for searching thought Big Data. We use Golay Code (Golay Code Clustering Hash Table "GCCHT") because of its time complexity of this method and the fuzziness aspect, although a lot of research has been done on clustering. According to, research and projects of Dr. Arai K. the computational time for one data set by using Fuzzy C-means is around 80 ms in average, so therefore, we cannot use Fuzzy C-means for big data, and clustering is a method to find alike data point [10, 17].

II. RELATED WORK

When taking a look at the history of application development, we find out that many applications were programmed with a fuzzy logic component in low of classical and zero-one logics because the latter two were incapable of representing many datasets and solving scientific problems. Creating an effective fuzzy searching algorithms with a fast time complexity has proven difficult for researchers. Some groups have faced obstacles in their approach to Fuzzy Clustering. In 2009, the problem lie in the inability to search more than one keyword. The implementation of prefix queries for multiple keyword searching consisted of multiple algorithms such as ranking answer highlighting results and utilizing synonyms [15].

In 2011, some research group works on the fuzzy keyword searching on encrypted cloud storage data with small indices [13], the problem of that method is time complexity of searching which is completely dependent on length of keyword and edit distance. In this paper we focus on the one mathematical model that was introduced in the year of 1949 by Golay, Marcel JE for digital coding [23]. In between 1979 and 1981, NASA[1] in project of deep space missions developed this algorithms as error correction technique by using hamming distance [16, 19] and in some research project in year of 1969, the main challenges of computer science committee was working on this statistical method as radio communication for Gilbert burst-error-correcting [12, 21], but that binary Golay Code works with 24 bits which is not perfect Golay coding algorithms and the best Golay algorithms which can run in linear time complexity is 23 bits [18]. This method is an implementation of Golay transformation in conjunction with 23 bits and allows for error

[1] National Aeronautics and Space Administration (NASA)

correction with two hashes utilizing the overlapping hash values [14]. In previous experiments, we concluded using the hash table is the most efficient method way for accessing data in constant time [13], Since 2014 we have been able to use the GCTHT in constructing a 23-bit meta-knowledge template for Big Data Discovery allowing for meta-feature extraction for clustering Structured and Unstructured Data (text-based and multimedia) [1, 2]. Yet the traditional use of GCTHT was hampering the ability to use fuzzy logic and we also realized the order of extracted meta-features was critical for Big Data clusterization. Re-arranging the order altered the results of the output ultimately requiring us to either use another algorithm to derive the best order or manually reorder to achieve best the best results [1, 2]. The other problems of that project is they use traditional way of GCTHT for their works and authors[1, 2] did not use the fuzzy logics of Golay Code, and another problem of these methods is the order of the features is critical for clustering the big data that means if we change the order of the features, the result will be changed in our output, so they need another algorithm for finding the best order of features or even they need to change the order of feature manually to find out best results. Intelligent Software Defined Storage is another example of usage of Golay Code Transformation Table [3].

III. METHODOLOGY

The technique we use in this paper is divided to 3 main part; first, we need to create the Golay Code Clustering Hash Table (GCCHT) where we don't go through it because it's implemented before [7, 24] , The Golay Code Clustering Hash Table data structure uses hash by 2^{23} indices and each index could have six 12-bits labels or one 12-bit label where 86.47 percent of indices of GCCHT has six labels and the rest has only one label . After The GCCHT is generated, we need to generate the mapping hash table or Golay Code Transformation Hash Table [18]. Because GCTHT is generated by GCCHT it has same data structure we 86.47 percent by 15 23-bit value and the rest has one 23-bit value.

The mapping of GCTHT is not efficient for searching because 13.53 percent of this table has only one 23-bit value and if we apply this algorithm for searching thought big data, absolutely we don't have accuracy. So we need to generate an efficient method of dictionary by using GCTHT. The generation of FuzzyFind Dictionary also has same data structure as GCTHT, but more than 98 percent of this hash table has 16 23-bit addresses and less than 2 percent has only one address.

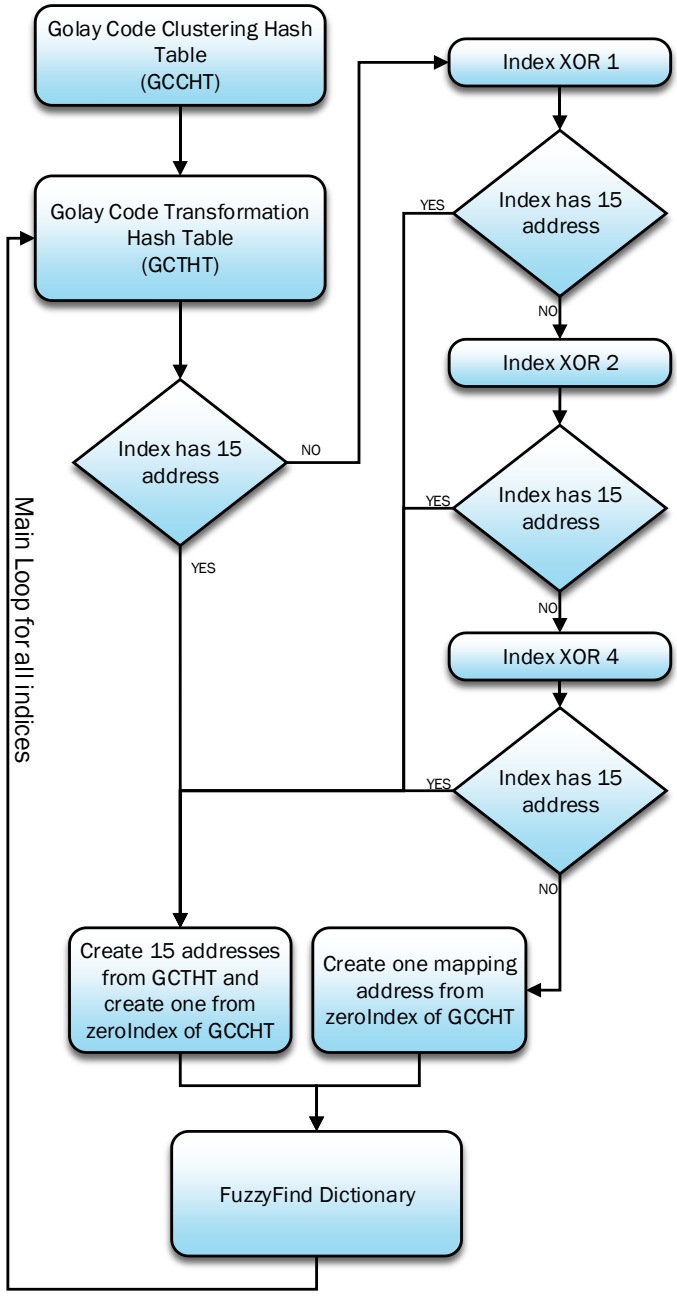


Fig. 1. Pipeline of generating FuzzyFind Dictionary using Golay Code Clustering Hash Table (GCCHT) and indicates the generating Golay Code Transformation Hash Table (GCTHT), and finally create FuzzyFind Dictionary, "This algorithms is pre-processing algorithm for text manning and searching through big data"

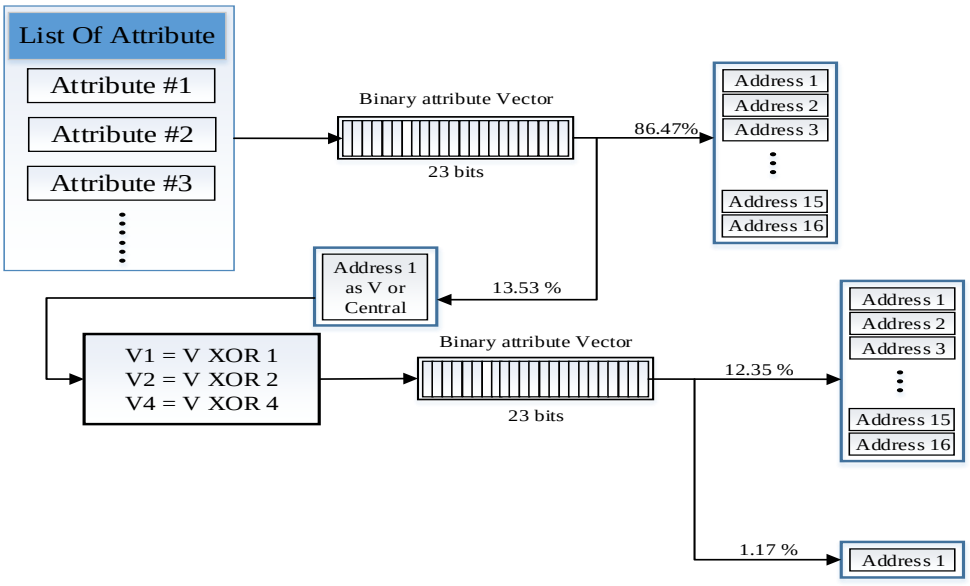


Fig. 2. General view of FuzzyFind Dictionary and indicates how improve the fuzziness logics from GCTHT [4, 7]

A. Golay Code Transformation Hash Table

In the Big Data environment an explicit utilization of all available information is not feasible. Resolution of this situation requires formation of knowledge that would render a substantial part of data as "less interesting" [20]. Human reaction to the Big Data avalanche is bounded rationality - a decision-making process complying with cognition limitations and imposed deadlines. The ideology of bounded rationality leads to a computational model of the brain that goes beyond the traditional Turing algorithmic revealing unconsciousness as the basis for sophistication [6].

Golay Code Transformation Hash Table (GCTHT) is hash table with 2^{23} indices from 0 to $2^{23} - 1$ record and 15 addresses items that are created by Golay Code fuzzy clustering table [18]. The GCTHT is created by Golay Code Clustering Hash Table (GCCHT) that include from 0 to 2^{23}-1 that contains 86.47 percent of them has six indices (labeled as fuzziness method) and 13.53 percent only has one indices (classical clustering, each data point belongs to one label) which means only 86.47 peents use real fuzzy logics clustering method and the rest following traditional clustering methods. GCTHT is created by GCCHT by following algorithms. GCTHT is created by all combination of labels.

$$\frac{6!}{4!\text{x } 2!} = 15 \quad (1)$$

$$|C| \sum_{k=0}^{e} \binom{n}{k} \leq 2^n \quad (2)$$

$$2^{12} \sum_{k=0}^{n} \left(\binom{23}{0} + \binom{23}{1} + \binom{23}{2} + \binom{23}{3} \right) = 2^{23} \quad (3)$$

B. Fuzzy Searching

Statistical data search engine, fuzzy Searching algorithms, when we can say that Fuzzy search algorithms are gained, modelling are designed by unstructured data from mathematical fuzziness modeling. The Fuzzy-Go search engine can thus automatically retrieve web pages that contain synonyms or terms similar to keywords, Fig. 4 [13].

Algorithm 1: Generating GCTHT by GCCHT

1. ***loop*** *i = 0 to 6*
2. ***loop*** *j = i+1 to 6*
3. *shift 12 bit of index one*
4. *HashPair = shifted hash1 OR hash 2*
5. *Shift one bit to right of HashPair*
6. *Shifted HashPair XOR HashPair*
7. ***end*** *of Loop 1*
8. ***end*** *of Loop 2*

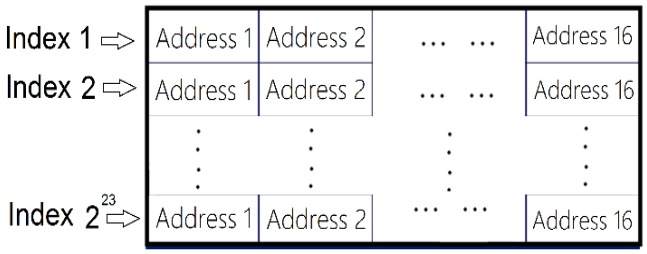


Fig. 3. FuzzyFind Dictionary data Structure is presented in this figure by use Hash Table

Algorithm 2: Generating Golay Code Clustering Hash Table

```
1.  loop  i = 0 to 2^23 //loop 1
2.      loop 1 to 24 //loop 2
3.          transform = (1 << i) & MASK_23; // Mask 23 bit all bits are 1
4.          codewordB = codeword  XOR  transform;
5.          recd = codewordB;
6.          recd = recd XOR decoding_table[get_syndrome(recd)];
7.          hash [i] = recd >> 11;
8.      end of loop 2
9.      Save Zero-Index (Zero index is needed for FuzzyFind Dictionary to create one address)
10.     if Hash elements have 2 different values Then
11.     Sort The Hash with 24 elements (It has 6 different values)
12.     Save in Golay Code hashes by 6 indices
13.     end of if
14.     if Hashes elements do not have 2 different values then
15.         Save NO-Value as first elements and save 1-index as Second Value
16. end of loop 1
```

2

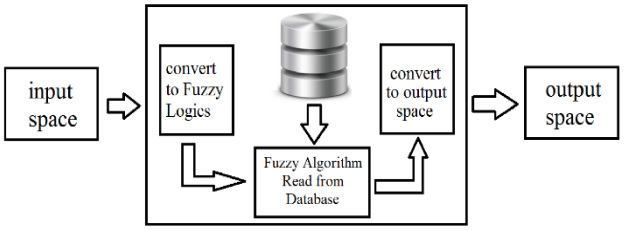


Fig. 4. This figure indicates how Fuzzy Search Engine works with input space and output space

C. FuzzyFind Data Structure

The FuzzyFind Search data structure works hash 2^{23} indices (0 to 8,388,607) and each indexes hash 16 different address as 23 bits from (0 to 8,388,607). The FuzzyFind Dictionary Data Structure contains by 16 addresses, but as regarding to 0.16 percent of this tables has only one address in mapping hash table. Construction of a dictionary is one of the basic undertakings in data structure developments. A dictionary is in essence a table that takes a key as an entry and if the key is present in the table returns some information associated with this key [20].

D. FuzzyFind Dictionary

Nowadays, fuzzy logic is used by a variety of search engines and cloud computing. Keyword based search methods allow us to select the retrieve files or words and has also been widely applied in plaintext search scenarios [22], such as Google or Bing search engine. The hash transformation utilizes the Golay Code. the decoding procedure which takes neighborhood spheres of radius 1surrounding the 23-bit binary vectors and yields hash codes as 12-bit keys: six hashes in 86.47%[7] of the cases(let's call it Case A) and one hash in 13.53% of the cases(let's call it Case B). In Case A, we apply Golay into a 23-bit vector to obtain 6 indices at 86.47% and further apply a transformation of the six 12-bit indices into 15 pairs. For Case B we apply Golay Coding into a 23-bit vector as we did with Case A, but will obtain 1-index which contains 13.53% of whole the GCTHT dividing it into two different categories. We apply Hamming Distances (HD) of 1 and 2 by yield and follow Algorithm 3. The Hamming distance between two codewords is equal to the number of bits in which they differ. It shows that if the error code and other code wants to become a few bits that must be changed to make this conversion is done without the error of the systems, brought to account. Hamming distance between two strings is equal to the length of information theory where the corresponding symbols are different. In other words, the minimum number of alternatives that will change one string to another string, or the number of errors that will convert a string to another string [22].

Algorithm 3: Generating FuzzyFind Dictionary using GCTHAT

```
1.  loop i = 0  to 2^23 – 1
2.    if  it does not have 15 mapping addresses Then
3.      index XOR 1
4.        if  it does not have 15 mapping addresses Then
5.            Index XOR 2
6.            if  it does not have 15 mapping addresses Then
7.                Index XOR 4
8.            if  it does not have 15 mapping addresses Then
9.                Create one addresses with one label
10.         else of if 3
11.               Create one address with one label and 15 addresses from mapping
12.       else of if 2
13.           Create one address with one label and 15 addresses from mapping
14.   else of if 1
15.           Create one address with one label and 15 addresses from mapping
16.   end of Loop
```

1000 → 00000000000001111101000

480 → 00000000000000111100000

1000 →		480 →	
[0]	5244416	[0]	256043
[1]	202860	[1]	43
[2]	202947	[2]	108
[3]	203166	[3]	199
[4]	204288	[4]	1920
[5]	202496	[5]	3840
[6]	442563	[6]	176236
[7]	442782	[7]	176327
[8]	443904	[8]	178048
[9]	446208	[9]	179968
[10]	799134	[10]	442567
[11]	800256	[11]	444288
[12]	802560	[12]	446208
[13]	1697280	[13]	819072
[14]	1699584	[14]	816896
[15]	6295296	[15]	7868160

Fig. 5. Sample result of FuzzyFind Dictionary and shows that two indices with hamming distance of less or equal than two has at least one same mapping index

IV. RESULTS

For testing FuzzyFind Dictionary we needs to test all cases which means Case A and B with 16 addresses and Case C with only one address. In our test algorithm we need to select one or more random word and create and look at the addresses of the theses word, after that find all hamming distance of this index within less and equal to 2 which means HD with 0, 1, and 2. As regarding to 23 bits indices we have 253 results for each indices with HD of 2 and 23 results for HD of 1 and only one index for HD is equal to 0, so the test algorithm needs to consider to 277 results [7]. After finding this indices, the algorithms must find at least one same address in the FuzzyFind Dictionary, all of them should have same address with our index which we choose. If the test algorithm find same address for all 277 indices, our FuzzyFind Dictionary is working, but even one of the test indices does not have same address with index we choose, The FuzzyFind Dictionary is not working.

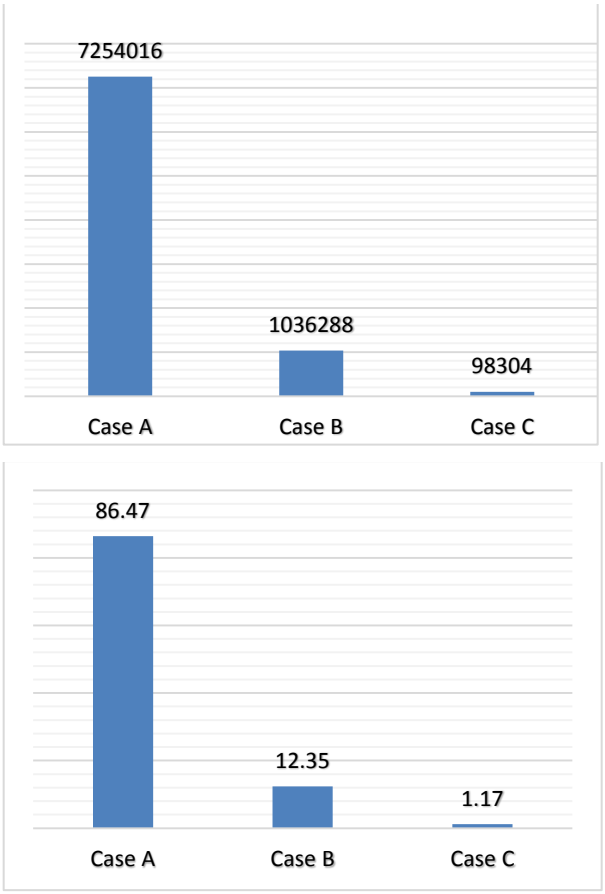


Fig. 6. Number of Case A, Case B and Case C where Case A is the number of indices which has 15 addresses into GCTHT, Case B is number of indices which has 1 indices but we can find 15 addresses by using Algorithm 1, and Case C indicates the number of indices we could not find any 15 addresses by using algorithm and only we create 1 addresses from Zeroindex of GCCHT, Chart B indicates Percentage of each group

V. DISCUSSION

The result of Case A is about 86.47 percent which has 16 different addresses and Case B that has one 12 bit hash but only can find 16 addresses by using V1, V2 and V4 (one 12-bit hash XOR with 1, 2 and 4) and algorithm can find 16 addresses is around 12.35 percent (Fig. 2), so by adding this two percent we can find out 98.83 percent of our FuzzyFind Dictionary has 16 addresses and less than 1.16 percent only has 1 address. In our implementation, called FuzzyFind Dictionary, we simply mapped the 26 letters of the English alphabet into 23 bits, reflecting the presence or absence of particular letters. This representation preserves closeness of word distortions in terms of closeness of the created binary vectors. Within Hamming distance 2 deviation. A hash transformation using the Golay code decoding procedure is applied to neighborhood spheres of radius 1 surrounding 23-bit binary vector [7]. We assume that you know already about the Golay Coding transformation. There are three basic provisions in realization of fuzzy retrieval with the suggested data structure (FuzzyFind Dictionary):

1) Attributes of information items have to be mapped to a binary vector in such a way that closeness in attribute discrepancies is translated into closeness of binary vectors in Hamming's metric,

2) The format of the fault-tolerant indexing based on this scheme imposes limitations on the length of the binary vector, in the case of Golay code the length is 23, and

3) The retrieved binary vectors cannot deviate from the search vector more than a relatively small value of the Hamming distance, typically by 2.

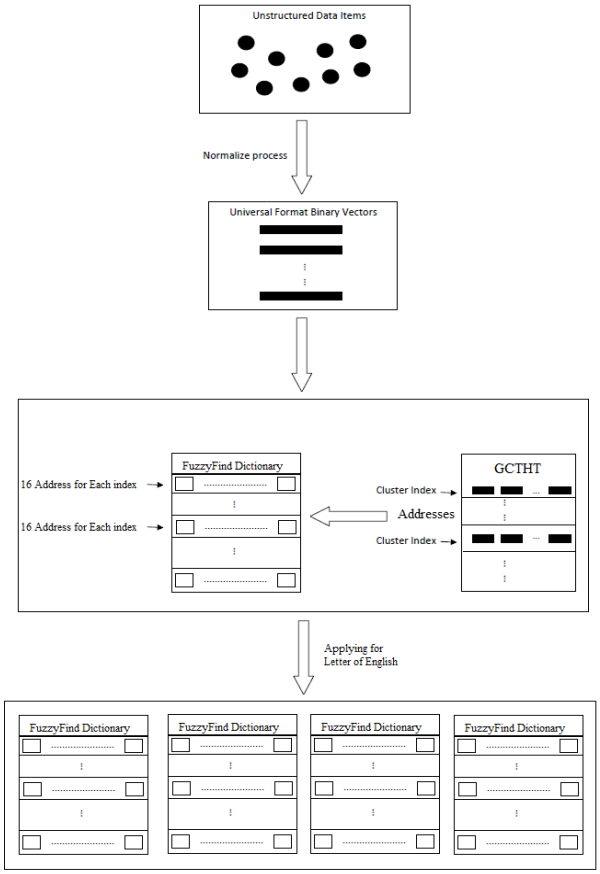


Fig. 7. This figure represent the fact that how we can use FuzzyFind Dictionary for more than 23 bits

VI. Conclusion and Future Work

Perhaps, all of the information provided in this paper research, might reflect the fact that generating FuzzyFind Dictionary by using Golay coding transformation Hash Tables is one of the efficient method of creating Fuzzy Dictionary for searching through big data. The Golay Coding Transformation Hash Table (GCTHT) and Golay Coding Address Tables (GCAT) works with linear time complexity and after creating GCTHT just we need to apply the into FFD algorithms by linear time complexity by following steps: First GCTHT is generated into hash table, so the algorithms has access by O(1) time complexity; second ,Generating FFD for all 2 power 23 needs linear time complexity as regarding to: Case A: They have six unique 12-bit indices and use 15 addresses of from GCAT and create another address with zero-indexes hash table. Case B: They have one 12-bit index, and one address, so we could find 15 addresses by the nearest data set by XOR 1, 2 or 4 and find 15 addresses by yield. Case C: these part is same with Case B, but FFD has only one address because of nearest data point which XOR with 1, 2 and 4 also has one address. This method has good time complexity for generating FuzzyFind Dictionary which is Linear, O(n) and assess to the FFD is constant time complexity O(1) because FFD used hash table. In this paper we show that a FuzzyFind Dictionary improved percentage of indices with sixteen addresses by 98.83 from GCTHT with 86.47 percent with 15 addresses. This FuzzyFind Dictionary can be used into search engine and also this method can be used in error correction of miss typing and sudden interruption of communication, loss, or lack of landmarks, fields containing Null, abbreviations unusual or abnormal fields are experiencing any reason. We have plan to use FuzzyFind Dictionary for indexing by supervised learning method and using historical data points. We have plan to distribute the source code and improve it for searching application in near future. Also this algorithms can be useful for mining gene sequences because the main challenge of bioinformatics, biology, and biological scientist is RNA-seq mining datasets [5]. We have plan to implement Fuzzy logics for DNA and RNA analyses which can be useful for diagnosis Tumor [9] as FuzzyFind logics.

Acknowledgment

We would like to sincerely thank Professor Simon Y. Berkovich for his guidance and a special thanks to our all faculty and staff of Department of Computer Science of The school of Engineering and applied Science at the George Washington University for their support.

References

[1] N. Bari, R. Vichr, K. Kowsari and S. Y. Berkovich. Novel metaknowledge-based processing technique for multimedia big data clustering challenges. The IEEE International Conference on Multimedia Big Data 2015.

[2] N. Bari, R. Vichr, K. Kowsari and S. Berkovich. 23-bit metaknowledge template towards big data knowledge discovery and management. IEEE International Conference on Data Science and Advanced Analytics 2014.

[3] S. Berkovich. Intelligent software defined storage. Presented at Proceedings of the 5th International Conference on Computing for Geospatial Research & Application, Washington, DC. 2014, .

[4] Kowsari, Kamran. Investigation of FuzzyFind Searching with Golay Code Transformations. Diss. M. Sc. Thesis, The George Washington University, Department of Computer Science, 2014.

[5] P. Mudvari, M. Movassagh, K. Kowsari, A. Seyfi, M. Kokkinaki, N. J. Edwards, N. Golestaneh and A. Horvath. SNPlice: Variants that modulate intron-retention from RNA-sequencing data. Bioinformatics Oxford 2014.

[6] R. E. Pino. Network Science and Cybersecurity 2014.

[7] Yammahi, Maryam and Kowsari, Kamran and Shen, Chen and Berkovich, and Simon. An efficient technique for searching very large files with fuzzy criteria using the pigeonhole principle. Presented at Computing for Geospatial Research and Application (COM. Geo), 2014 Fifth International Conference. 2014, .

[8] K. Kowsari. Comparison three methods of clustering: K-means, spectral clustering and hierarchical clustering. arXiv Preprint arXiv:1312.6117 2013.

[9] P. Mudvari, K. Kowsari, C. Cole, R. Mazumder and A. Horvath. Extraction of molecular features through exome to transcriptome alignment. Journal of Metabolomics and Systems Biology 1(1), 2013.

[10] K. Arai. Comparative study between the proposed GA based ISODAT clustering and the conventional clustering methods. International Journal of Advanced Computer Science and Applications (IJACSA) 3(7), 2012.

[11] S. Berkovich and D. Liao. On clusterization of big data streams. Presented at Proceedings of the 3rd International Conference on Computing for Geospatial Research and Applications. 2012, .

[12] H. Yu, T. Jing, D. Chen and S. Y. Berkovich. Golay code clustering for mobility behavior similarity classification in pocket switched networks. J.of Communication and Computer, USA (4), 2012.

[13] C. Liu, L. Zhu, L. Li and Y. Tan. Fuzzy keyword search on encrypted cloud storage data with small index. Presented at Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference On. 2011, .

[14] H. Yu. Golay code clustering using double golay encoding technique. 2011.

[15] S. Ji, G. Li, C. Li and J. Feng. Efficient interactive fuzzy keyword search. Presented at Proceedings of the 18th International Conference on World Wide Web. 2009, .

[16] J. West. Commercializing open science: Deep space communications as the lead market for shannon theory, 1960–73. Journal of Management Studies 45(8), pp. 1506-1532. 2008.

[17] K. Arai and A. R. Barakbah. Hierarchical K-means: An algorithm for centroids initialization for K-means. Reports of the Faculty of Science and Engineering 36(1), pp. 25-31. 2007.

[18] E. Berkovich. Method of and System for Searching a Data Dictionary with Fault Tolerant Indexing 2007.

[19] T. M. Thompson. From Error-Correcting Codes through Sphere Packings to Simple Groups 2004(21).

[20] S. Berkovich, E. Berkovich, B. Beroukhim and G. Lapir. Organization of automatic spelling correction: Towards the design of intelligent information retrieval systems. Presented at The 21st National Conference of the ASEM, Washington, DC. 2000, .

[21] L. Bahl and R. Chien. On gilbert burst-error-correcting codes (corresp.). Information Theory, IEEE Transactions On 15(3), pp. 431-433. 1969.

[22] G. Forney Jr. Generalized minimum distance decoding. Information Theory, IEEE Transactions On 12(2), pp. 125-131. 1966.

[23] M. J. Golay. Notes on digital coding. Proceedings of the Institute of Radio Engineers 37(6), pp. 657-657. 1949.

[24] F. Alsaby, K. Alnowaiser and S. Berkovich. Golay code transformations for ensemble clustering in application to medical diagnostics

AUTHORS PROFILE

Kamran Kowsari, He is a Ph.D Candidate in Computer Science at The George Washington University Washington DC USA. He received his Master of Science in Computer Science, Data Mining and Machine Learning at The George Washington University Washington DC USA. He has more than 6 year's experiences in software development, system and database engineering experience, and research. His experience includes numerous projects and academic projects. He is interested and has experience in Machine learning, mathematical modeling, algorithms and data structure, Real-time rendering use machine learning, and data manning, Computer Graphics and visualization, volumetric Rendering .

Maryam Yammahi, She received her Ph.D degree in Computer Science at The George Washington University, her focus area is Data manning and Machine Learning, Fuzziness Searching, mathematical model.

Nima Bari, She received her Ph.D degree in Computer Science at The George Washington University, her focus area is Data manning and Machine Learning, Fuzziness clustering, meta-knowledge and knowledge discovery.

Roman Vichr, He received his PhD. degree in Material Engineering from the Institute of Chemical Technology of Prague in 1992. He has more than 20 year's experiences in international software development, system and database engineering experience. His experience includes numerous projects for Fortune 100 companies and government bodies at the federal and state level.

Faisal Alsaby, He is currently a Ph.D candidate at the GWU majoring in Computer Science. He received an MS degree in Computer Science from the George Washington University, Washington, DC, and USA in 2012. His research interests are big data clustering algorithms, machine learning.

Simon Y. Berkovich, He received his Ph.D in Computer Science, 1964, Institute of Precision Mechanics and Computer Technology, USSR Academy of Sciences, his MS in Applied Physics, 1960, Moscow Physico-Technological Institute, He is Professor and Faculty at School of Engineering and applied Science at The George Washington University. His major area of research is information retrieval, computer organization, and mathematical modeling. He has more than 100 technical publications and 30 patent

An Approach to Extend WSDL-Based Data Types Specification to Enhance Web Services Understandability

Fuad Alshraiedeh
Computer Science Department
Philadelphia University
Amman- Jordan

Samer Hanna
Software Engineering Department
Philadelphia University
Amman- Jordan

Raed Alazaidah
Computer Science Department
Philadelphia University
Amman- Jordan

***Abstract*—Web Services are important for integrating distributed heterogeneous applications. One of the problems that facing Web Services is the difficulty for a service provider to represent the datatype of the parameters of the operations provided by a Web service inside Web Service Description Language (WSDL). This problem will make it difficult for service requester to understand, reverse engineering, and also to decide if Web service is applicable to the required task of their application or not. This paper introduces an approach to extend Web service datatypes specifications inside WSDL in order to solve the aforementioned challenges. This approach is based on adding more description to the provided operations parameters datatypes and also simplified the WSDL document in new enrichment XML-Schema. The main contributions of this paper are:**

1. Comprehensive study of 33 datatypes in C# language, and how they are represented inside WSDL document.

2. Classification of the previous mentioned datatypes into 3 categories: (Clear, Indistinguishable, and Unclear) datatypes.

3. Enhance the representation of 18% of C# datatypes that are not supported by XML by producing a new simple enrichment XML-based schema.

4. Enhance Web Service Understandability by simplifying WSDL document through producing summarized new simple enrichment schema.

Keywords—Datatypes; Understandability; Web Service

I. Introduction: What are Web Services?

A review of the various studies showed that a large number of definitions for Web Service have been proposed. For example [1] defined the Web Service as software components that allow access to functionality via a Web interface network. Additionally, [2] and [3] defined it as a software system designed to support machine-to-machine interaction over a network. These brief definitions detail a new breed of Web applications with self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. According to this paper, Web Services are defined as a collection of applications (interface application) or a collection of systems (endpoints) interacting with each other by exchanging data and information over networks. Each service has its self-located, self-describing and also self-operational properties. If one of these endpoints is to provide service over network (Internet or intranet), then the provider must publish a full and detailed explanation for this service. This detailed explanation is called Web Service Description Language (WSDL) [1][4]. WSDL makes it easier for other endpoints which share the same network to know more about the provided service, and then to decide if this service is applicable for their needs or not

However, Web Service faces numerous challenges and problems, including, but not limited to the following [5]:

a) The trustworthiness problem: The Service Requester can only see the contract (WSDL) of a Web Service but not the source code. This fact has caused Service Requesters to question the trustworthiness of Web Service because Service Requesters do not trust Web Services that were implemented by others without seeing the source code. [6] mentioned that this problem is limiting the growth of Web Service applications and that these applications will not grow unless researchers meet this trustworthiness challenge. [7] stated that the current methods and technologies cannot ensure Web Service trustworthiness and that for Web Services to grow, researchers must address this challenge.

b) ***Vulnerability to invalid inputs by malicious Service Requesters****: Since Web Services are advertised in the Internet, any Service Requester can access this Web Service and some of these might be malicious Requesters that aim to do harm. The Web Input manipulation vulnerability is 59.16% of the overall Web Services vulnerabilities[8] and that is why Web Services should be tested against this kind of fault to assess if a Web service is vulnerable to input manipulation attacks in order to increase Web service trustworthiness. [9] mentioned that testing that a program does what it is supposed to do is only half the battle, the other half is to test whether the program does what is not supposed to do. In other words, to check if a program is vulnerable to invalid input.*

In this paper, we have investigated the problem of the Web Services understanding, and how to distinguish between the input/output parameters datatypes for the Web Service operations. The result of our research is a tool and its algorithm for extending the XML-Schema to represent the

Web Services operations parameters datatypes to reach better comprehension for the Web Service functionality. Unlike previous approaches, which give a semantic for Web Services during its WSDL documents, ignoring uncertainty of operations parameters datatypes declarations, our proposed approach can analyze all the operations of the Web Service and then classify the input/output parameters that operations need.

Section two discuses related work of web service understandability. In section three we propose the model while in section four brief discussion about the proposed tool. The last section introduces the conclusion and future work.

II. Related Work: Web Service Understandability

As it is well known Web Services include a large number of research fields, many studies and researches have been published in the Web Service area. For example, if we take a sample of these studies, we note that some of the researches focused attention on how to build a Web Service. Other researchers proposed approaches for specifying semantic Web Services composition using UML (Unified Modeling Language) profile [10][11]. Other researchers recommend a model-driven process for web services development [8], and there are many others.

A. Overview

Many recent studies have been published in the field of Web Service. The goal of these studies is to facilitate and increase the communication among the distributed systems, and also use Web Service reverse engineering to facilitate the reuse and composition of the Web Services [12], to ultimately facilitate the exchange of information over the networks. In order to facilitate the understandability of the Web Services functionality and what these Services are offering to its requester [11] a way must be found to facilitate the description of Web Services.

While reviewing several previous studies concerning the field of Web Services, obviously it was necessary for all researchers to mention several main concepts, such as XML, UML, SOAP, XSD and also WSDL [13], which in turn are used to perform selections, descriptions, discovery, composition, and interaction with the Web Services [14]. All researches attempted to built a bridge between the Web Services providers and Web Services requesters to reach better comprehension for Web Service functionality from the Web Service requester to increase the exchangeability of information between heterogeneous applications.

The related research to this paper is about representing the information inside WSDL in a more understandable form.

In this paper, we classify the related Web Service understandability into several aspects and we also give a brief overview for each research and the limitations for each research which we will try to solve in this paper.

B. Reverse Engineering Approach for Semantic Web Services Composition

Reference [12] presented an approach to facilitate and raise the degree of automation for Composition of Web Services. The approach used UML-Model to give graphical description for the Composition Web Services, The proposed approach summarized in three steps of Web Services Composition:

1) Using RE methodology to turn the selected Web Services WSDL documents to one or more UML-Model depending on the number of selected services.

2) Integrating these models into one UML-Model which implements all integrated UMLs using one of the UML-tools.

3) In step 2 a new Web Service is created (Composition Web Service) and by using one of the UML-tool a new description for this Web Service is created which called OWL (Web Ontology Language).

Here we can criticize this work in simple terms. The final description OWL is dependent on UML-Models. These models are created using different tools and also may implement heterogeneous applications. Suppose one or more of these applications is used by one of the Datatypes not implemented clearly in WSDL, such as char, array, array of objects. Here, the model which implements the Web Service before composition will have ambiguity, but after it composes with others, inevitably the ambiguity will increase, so that this approach is good and will work properly if all of the datatypes of Web Services parameters are represented clearly. If one or more parameters are represented ambiguously, surely it will face missed understanding for Web Services requesters and developers. Our proposed approach seeks to overcome these challenges and also to reach batter comprehension for Web Service functionality.

C. Model-Driven Web Service Development

In this field [15] has been proposed another way to give more comprehension for the Web Service description to make it easy for a requester to decide if the selected Web Service is applicable for his requirement or not.

The proposed approach summarized in three steps of Model-Driven Web Service Development which divided into following steps:

1) The WSDL are converted to graphical modeling language (UML).

2) Integrate with other UMLs for a composition Web Service.

3) A new Web Service descriptions are exported.

This approach is not different from the previous one but it added a Pure UML modeling strategy supported by implementation of two-way conversion rules from WSDL to UML and also from UML to WSDL documents. But this rule does not avoid the issues and problems which we are trying to solve, so the same challenges of understandability are still present.

D. Reverse Engineering Existing Web Service Applications

One of other approaches which proposed to deal with Web Services description is MIDAS-CASE. It aimed to extend the UML language to support the modeling of the Web Services description, and then automatic generation of the WSDL document for concerned Web Service [16].

As we illustrated before, a WSDL file is an independent XML-based standard which is proposed by W3C to represent the Web Services functionality [16]. One of the other properties for WSDL is that it is XML-based version of Interface Definition Language (IDL), so MIDAS-CASE is one of the framework methodologies that aimed to facilitate the development of Web Service Information System depending on Model Driven Architecture (MDA). MDA has three dimensions: CIM (Computation Independent Model), PIM (CIM Platform Independent Model), and finally PSM (Platform Specific Model), [16] used in his approach.

This approach is divided into three steps:

1) The client defines extended UML model which is then stored as XML-based.

2) Reference [16] defined an XML-Schema to describe Meta-Model for extended UML which was introduced in step one, and then WSDL document is automatic generated using one of the existing tools.

3) The XML document now becomes an instance of the XML-Schema. The XML document is not valid according to the XML-Schema, thus we conclude that the model is not valid, since the model does not carry out the meta model.

This MIDAS-CASE web service architecture is one of the most important approaches[17] used for a Web Service development, but as shown it doesn't deal with WSDL document which are automatically generated and also built according for extended UML model. UML models introduced and extended are vulnerable for the risks and challenges previously mentioned and never get clear understandability for Web Services functionality.

III. The Proposed Model: Extending the XML-Schema Datatypes Specification to Reach Better Comprehension of the WSDL

Based on the previous review of Web Service understanding we noted that no approach attempted to solve the inconsistency and ambiguity in defining the Web Service operations parameters datatypes.

1) Datatypes Description

The previous approaches solved the problems of the Web Services understanding, reusing and comprehension by using UML to give graphical definitions for Web Services and its functionality. These approaches have several advantages such as :

a) The graphical implementation gives a full summary for the Web Service functionality but with few details.

b) The graphical implementation is easier to understand than textual implementation (WSDL document).

c) Can be easily understood even by non-specialists in Web services.

However the graphical implementation ignores the most important part which is the needed data that must be used to bind with Web Services, on which we are focusing in this paper. As motioned in chapter two, messages are used to bind with Web Services by filing an application which published by the Web Service provider with parameters. Surely these parameters must clearly appear to the users without ambiguity; because any error in the filling of these parameters will lead to Web Service failure which we always seek to ensure does not happen. Therefore we are proposing an approach based on extending the XML-Schema to reach better comprehension and reusing the Web Services functionality, which in turn leads users to understand all Web Services operations and also to determine all the parameters datatypes which Web Services need.The proposed approach is accompanied with a tool in order to prove the approaches usefulness and compare it with other approaches. The tool can be auto run when the user tries to bind with the Web Service. This tool can answer the major question of this paper, that is: Can we extend the XML Schema datatypes to reach for a better comprehension of the WSDL documents by the service requester and provider of Web Services? Other questions could be inspired from the previous major question, such as:

a) Do all of the parameters datatypes need to be extended ?

b) Can the tool distinguish between the parameters datatypes ?

c) How can we reach better comprehension for the Web Services ?

Section 3.2 shows the model which this paper proposes to solves the datatypes description problems, and we used several datatypes as case studies to illustrate all model steps. These datatypes can be divided into three categories:

a) Clear Datatypes .

b) Indistinguishable Datatypes .

c) Unclear Datatypes .

Figure 1 shows how the tool deals with these datatypes.

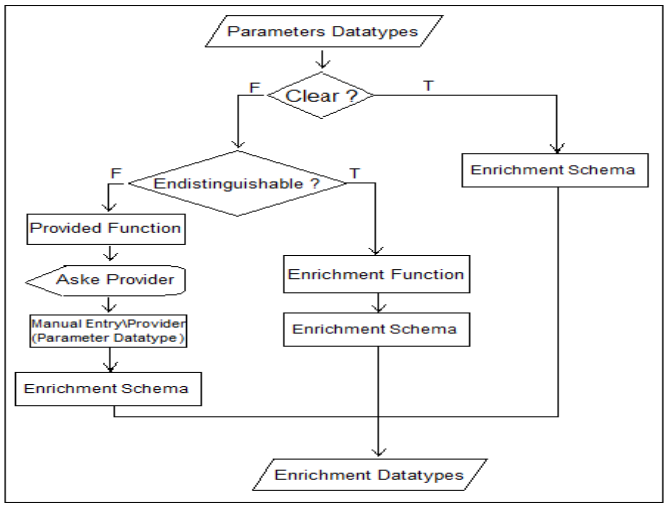


Fig. 1. The datatypes processing

2) The Proposed Model

The proposed model consists of five phases: a) extract the WSDL document, b) extract datatype specification, c) add more description and annotation, d) add constraining facets to the datatypes, and e) extract UML class diagram using any published tool. Figure 2 shows the general structure of the proposed model.

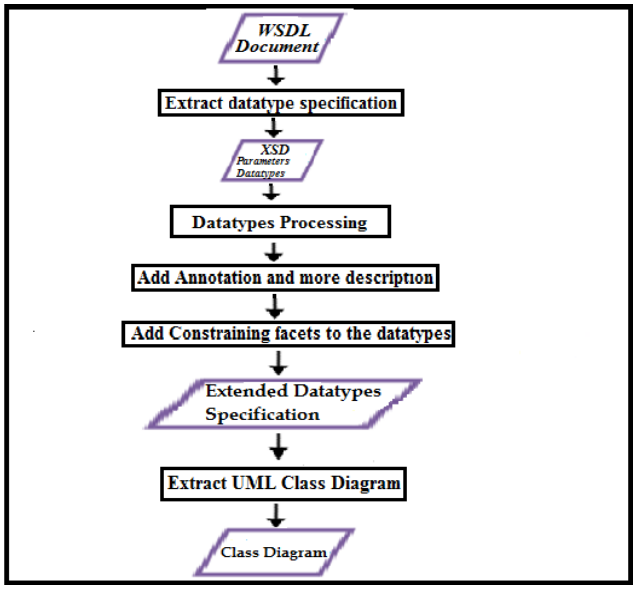


Fig. 2. The proposed model

The input of proposed approach is a WSDL document. WSDL description document is complex in nature, usually automatically generated by one of the Web Services development tools such as .NET, Apache Axis, Java etc.[15]. Each of these tools has its own particular way to define or to implement the input and output parameters datatypes for each of the Web Services operations. Here we are attempting to overcome these differences. The first step addresses the question of how to extract the WSDL document. WSDL documents are compulsorily published with Web Service; the provider cannot publish his own service application until its description (WSDL) generated, so that any developer or user wanting to know more about the operations or services then he can review the provided WSDL document. There are many ways to extract WSDL document, but here we are looking to make our proposed tool to run automatically when the Web Service client, user, and also developer want to bind with the Web Service and in the final stage give him a clear and simple description for Web Service input/output parameters datatypes. The proposed tool extracts the WSDL document and then extracts the XSD. Then the tool can distinguish between the input/output parameters datatypes which may need more description and constraints with which do not need.

First we discuss how different .NET , Java Datatypes are specified inside WSDL, given that WSDL documents depends on the XML Schema Data types (XSD) system, datatype specification produced when using an Axis2 based tool to build a Web Services is also compared. Suppose we have a method with *byte* and *char* Datatypes parameters, the question here is: how do these parameters will be implemented inside WSDL using the aforementioned platforms?. Next examples will illustrate that.

1- byte Data Type

a) The byte in C# :

TABLE I. *BYTE* DATA TYPE AND ITS ALIAS (BYTE)

Short Name	C# Class	Type
Byte	Byte	Unsigned integer

XSD Equivalence of C# *byte* :

byte or its alias *Byte* are equivalent to *unsignedByte* in XSD

Example 1

For the following method

byte byteExample(Byte bytepar)

The input and output parameters datatypes of previous method are specified by XSD inside WSDL as

a. <xs:element type="xs:*unsignedByte*" name="bytepar" >.
b. <xs:elementtype="xs:*unsignedByte*" name="byteExampleResult">.

b) byte in Java :

TABLE II. *BYTE* DATA TYPE AND ITS WRAPPER (*BYTE*)

Name	Wrapper Class	Type
Byte	Byte	Signed integer

XSD Equivalence of Java *byte*

byte or its Wrapper class *Byte* are equivalent to *byte* in XSD.

Example 2

For the following method

byte byteExample(Byte bytepar)

The input and output datatypes for above method are specified by XSD inside WSDL:

a. <xs:element type="xs:*byte*" name="arg0" minOccurs="0"/>.
b. <xs:element type="xs:*byte*" name="return"/>.

c) Axis2 Equivalence of byte

Example 3

For the following method

byte byteExample(Byte bytepar).

The input and output datatypes are specified by XSD inside WSDL as:

a. <element name=*"bytePar"*type=*"xsd:byte"*/>
b. <element name=*"byteExampleReturn"* type=*"xsd:byte"*/>

2- *char* datatype

a) char in .NET

TABLE III. *CHAR* DATA TYPE AND ITS ALIAS

ShortName	.NET Class	Type
char	Char	A single Unicode character

XSD Equivalence of .NET *char*

The WSDL document for the Web Service which used *char* or *Char* alias datatype as request and response operations is defined as custom datatype (ns : char) where (ns) is a .NET namespace. *char* or its alias *Char* defined inside WSDL as the following example:

Example 4

For the following method

public char charExample(Char charPar)

The *char* datatypes are specified by XSD inside WSDL as:

a. <xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/" minOccurs="0" name="charInput" type="**q1:char**"/>.
b. <xs:element xmlns:q2="http://schemas.microsoft.com/2003/10/Serialization/" minOccurs="0" name="charExampleResult" type="**q2:char**"/>

b) char in Java

TABLE IV. *CHAR* DATA TYPE AND ITS ALIAS (BYTE)

Name	Wrapper Class	Type
char	Character	A single Unicode character

XSD Equivalence of Java *char*

char or its Wrapper class *Character* are equivalent to *unsignedShort* in XSD

Example 5

For the following method

char charExample(Character charPar)

The char datatypes for previous method are specified by XSD insideWSDLas:

a. <xs:element name="arg0"type="**xs:unsignedShort**"minOccurs="0"/>.
b. <xs:element name="return" type="**xs:unsignedShort**"/>

Axis2

For the following method

char charExample(Character charPar)

The following warning was generated

The service class "wtp.Datatypes" does not comply with one or more requirements of the JAX-RPC 1.1 specification, and may not deploy or function correctly.

The method "charExample" on the service class "wtp.Datatypes" uses a data type, "char," that is not supported by the JAX-RPC specification. Instances of the type may not serialize or deserialize correctly. Loss of data or complete failure of the Web service may result, and the following datatype specification was generated inside WSDL.

<element name="charPar" type="**xsd:anyType**"/>

In this section, we illustrate the implementation differences between three tools, and we also show how each of these tools implement *byte, char* input output parameter datatypes. These differences create misunderstandings for the Web Services requesters, clients, users and also developers because these datatypes are not implemented in the same and formal way as we have seen in *char* datatype. But here in our paper we want to implement our proposed approach on .NET tool as case study.

3) Constraints Modification

In all of the previous examples we provided an illustration for parameters datatypes but other aspect will process by our proposed tool which is the constraints for these parameters, so in next paragraphs we illustrate this aspect.

We noted in previous figures, WSDL contain the **< xs:element.. minOccurs="0" name ...>** part, it contains ***minOccur="0"*** and also in other case may contain ***maxOccur*** =" ".

To illustrate this aspect we discussed some possible case studies. Figure 3 includes three examples of occurrences of a specific element.

<element name="one" type="string" **minOccurs="3" maxOccurs="4"**/>.
<element ref="target:one" **maxOccurs="10"/>.**
<element name="position" ""**minOccurs="0" maxOccurs="unbounded"**/>**.**

Fig. 3. min/max occurrence cases

In example 1 Figure 3 declares that element **<one>** should appear within the instance document a minimum of three time and a maximum of four times. Example 2 declares an element using a reference to global **<one>** declaration with maximum attribute with 10 time appearance. The last example specifies the element **<position>** which may not appear at all "**minOccurs="0"** and it may also appear for infinite number of times **maxOccurs="unbounded**" . The default value for each minimum and maximum is 1 time appearance, meaning if not specified by provider, then the element must be appear for one time at least. An additional constraint to which the provider must adhere when specifying min and max occurrence is that the max value must always be greater than or equal to the min value.

4) Proposed Tool Environment

The tool solves the understandability problem by the creation of new XML-Schema called enrichment schema. This schema, simplified as much as possible, consists of just the WSDL parts which the requesters need to know what the concerned Web Service serve. Chapter five consists of the

enrichment schema, the enrichment algorithm, interfaces for how our proposed tool runs, and also a table for 33 data types with examples for each type and the method to implement inside WSDL documents and how these data types are classified by our proposed tool.

5) Datatypes Classifications

In this section we show how the proposed tool can be distinguished between the datatypes and how it deals with these differences. In Figure 4 we show three groups for parameters datatypes; these types are included in three possible cases.

A. Clear Datatypes :

In this case, the datatypes are implemented in a formal way and the datatypes are implemented as it is without any changes, so there is no need for any enrichment. The new WSDL document generated by our proposed tool (Extended XML-Schema) will have the same XSD datatypes without any modification to the original WSDL document. The enrichment part will have the same implementation for the datatypes with no changes, as the datatypes are clear and need no annotations. We will show the enrichment schema in chapter five with more details. The next example shows how the .NET tool implements *double* datatypes as case study and also shows how the proposed tool deals with this case.

double datatype

1- *double* in C#

TABLE V. *DOUBLE* DATA TYPE

Short Name	.NET Class	Type
Double	Double	Double-precision floating point type

XSD Equivalence of C# double

double or its alias *Double* are equivalent to ***double*** in XSD.

Example 6

For the following method

public double doubleExample(Double doublePar)

This example for double datatype implementation shows how C# tool implements the double datatype inside WSDL document:

a. **<xs:element minOccurs="0" name="doublePar" type="xs:*double*"/>.**
b. **<xs:element minOccurs="0" name="doubleExampleResult" type="xs:*double*"/>**

The proposed tool will firstly extract the WSDL document and then extract the XSD part, and finally check if the datatype is clear or not. In this example the tool will skip the third and forth steps of our proposed model because there is no need for any annotations or constraints. The parameter (*doublePar*) is given its type *double* without any ambiguity. The following steps summarize how the tool functions:

Step 1: Extract the WSDL document for the (public ***double*** doubleExample(***Double*** doublePar)) method ,

Step 2: Extract the parameter datatypes XSD as:

a. **<xs:element .. type="xs:*double*"/>(**Input parameter).
b. **<xs:element ... type="xs:*double*"/>**(Output parameter).

In this phase the tool can be distinguished that these parameters is not needs for more description it is clear and the requester can know that it is *double* datatype as it is.

Step 3: No annotation to be added. The enrichment part will have the same implementation for datatype as it is in original WSDL document with no annotations.

Step 4: Add constraints that the elements *"doublePar"* and *"doubleExampleResult"* will appear one time or more for both as :

<enr_min_appear> "1" </enr_min_appear>
<enr_max_appear> "unbounded" </enr_max_appear>

Step 5: The class diagram will not be changed after we run our proposed tool because the WSDL document has not changed. We will show how the proposed tool introduced the enrichment schema and also the enrichment algorithm for the three classification datatypes in chapter five.

1) Indistinguishable datatypes

Here other cases of datatypes are discussed. In example 5, we shows one of the datatypes which is clearly implemented inside WSDL document, but here we will show other cases of the datatype (class datatype as a case study).

The proposed tool can distinguish the mismatch defined, the WSDL document extracting then XSD extracting and then apply the third and fourth steps by adding more descriptions (annotations, constraints).

Classes datatype

Classes In C#

This sample of method code shows the implementation for player member which defined as class with two parameters, his name and his nickname both of parameters defined as *string* datatype.

```
[DataContract]
  public class Player
  {
    [DataMember]   public String Name1 { get; set; }
    [DataMember]   public String NickName {get; set;}
   }
```

The *string* datatypes are specified by XSD inside WSDL as:

a. <xs:element minOccurs="0" name="Name1" nillable="true" type="**xs:string**"/>
b. <xs:element minOccurs="0" name="NickName" nillable="true" type="**xs:string**"/>

This example for classes applied to one of the datatypes that can be distinguished by the proposed tool. The class here

(<xs:element name="Player" nillable="true" type="tns:Player" /> has two parameters and in other case may have more. Each of these parameters is defined in a separate line so that the proposed tool can determine that these parameters belong to the class datatype. The annotations and constraints will be added to the new enrichment XML-Schema to give more description than the original WSDL document.

Now we can show how our proposed tool deals with *uint* datatype .

Example

Public uint uintExample(UInt32 uintpar)

The unit datatype is implemented inside WSDL as:

a. <xs:element minOccurs="0" name="uintPar" type="**xs:unsignedInt**"/>
b. <xs:element minOccurs="0" name="uintExampleResult" type="xs:unsignedInt"/>

The uint datatypes is implemented inside WSDL as *unsignedint* which is a custom declaration for .NET, and it may be represented using other tool by other way. So this datatype process by our proposed tool as following.

Step 1: Extract the WSDL document for Web Service.

Step 2:Extract the parameter datatypes XSD as:

a. <xs:element ... type="xs:unsignedInt"/> (Input parameter).
b. <xs:element .. . type="xs:unsignedInt"/> (Output parameter).

Step 3: Add annotation for the proposed schema that provides the correct type for the input parameter " *uintPar* " and output parameter *"uintExampleResult"* is *uint* type for both as :

<enr_type > *"uint*" </enr_type>

Step 4: Add constraints that the input parameter *" uintPar "* and output parameter *" uintExampleResult "* will appear zero times or more for both as :

<enr_min_appear> "0" </enr_min_appear>

<enr_max_appear> "unbounded" </enr_max_appear>.

Step 5: Class diagram will be generated during any published tool depending on the new enrichment WSDL document.

2) Unclear Datatypes:

Here is the third classified datatype which cannot be addressed until back to the Web Service provider itself. The tool can execute step 1 and step 2 and then checking about the datatype classification. In the previous two classifications the tool can address the problem automatically; in unclear datatypes it stops and asks the Web Service provider about which datatypes the provider specified for Web Service operation parameter datatypes.

We discussed this case using Array and List as case study

Array and List Datatype:

Suppose the provider defined the following sample code of the Web Service

int[] ArrayExample(int[] arrayPar.

The two operations request and response for previous *array* of *integer* declaration are defined inside WSDL as :

a. 0/Serialization/Arrays" minOccurs="0" name="arrayPar" nillable="true" type="*q3:ArrayOfint*"/>
b. 0/Serialization/Arrays" minOccurs="0" name="ArrayExampleResult" nillable="true" type="*q4:ArrayOfint*"/>

On the other hand (*list* datatype) is defined as:

List<int> ListExample(List<String> listPaList<int>

The WSDL document declaration for the *(list* of *int*) and *(list* of *string*) is shown as:

a. 0/Serialization/Arrays" minOccurs="0" name="listPar" nillable="true" type="*q5:ArrayOfstring"/>*
b. /Serialization/Arrays" minOccurs="0" name="ListExampleResult" nillable="true" type="*q6:ArrayOfint"/*

Both of *list* and *array* are defined as the same way (*ArrayOfint*, *ArrayOfstring*), both of them are defined as array datatype. The question here is how may the user understand which type of data the operation needs, and how can the user distinguish between the array datatype and list datatype? So that the proposed model can answer these questions by referring to the service provider itself to determine the specific datatype, and then presenting it for a requester in a simple and clear way, the proposed tool functions for this case as follows:

Step 1: Extract the WSDL document for Web Service.

Step 2: Extract the parameters datatypes XSD as:

a. <*Serialization/Arrays... "q5:ArrayOfstring"/>*
b. <*Serialization/Arrays... q6:ArrayOfint "/>*

Step 3: Here the tool will back to Web Service provider by sending to him an message as interface, asking him to select from a datatypes list which datatype he given for the operation which written its name in the interface. After the provider select the parameter datatype then the tool can add the selected parameter datatype to the enrichment schema.

Step4: Add constraints that provide how many times the input and output parameters will appear as we presented before:

a. <enr_min_appear> " " </enr_min_appear>
b. <enr_max_appear> " " </enr_max_appear>.

Step 5: The new class diagram for enrichment schema will created the simplest and clearest way.

IV. TOOL ENVIRONMENT

1) Enrichment Algorithm

The tool runs automatically when a requester wants to bind with a Web Service doing its process. Finally a new enrichment WSDL document attached, and the requester can examine it.

This tool executes its functionality during the proposed enrichment algorithm, shown in Figure 4.

```
enr_type_algo
{
 Input_xsd
If type = (int, short, long, float, double, Boolean, decimal, string, timezone) then enr_schema(name, type, type)
elseif type = (ns*:type, datetime, anytype) then enr_schema OperationName(Prname, type, provided_type)
elseif enr_schema OperationName(ParName, type, enr_type_func)
}
* ns : Custom Datatype
```

Fig. 4. The proposed enrichment algorithm

This algorithm has three *if statements*. The first one which is the best case of the proposed algorithm gives its output enrichment WSDL document without any *function* call or communications needs ,while the second *if statement* needs to call a function *(provided_type_func)*.

This function returns back to the Web service provider to get the parameter datatype needed for the specific operation and this could be a small drawback of the proposed algorithm since it needs some communications with the providers and it may cause to increase the runtime of the proposed algorithm. This function structure is shown in Figure 5.

```
function provided_type_func(string)
 {
  if type = undefined* then
  messagbox contains
  {   "Please select datatype the parameter datatype"
{
       Listofdatatypes
      {RadioButton}
      {SubmitButton}
*  unclear datatype}
```

Fig. 5. Provided Datatype Function

The last function which appears in Figure 6 is *(enr_type_func)*. By this function the proposed tool can determine which datatype the provider selected for the specific parameter, as shown in Figure 6.

```
function enr_type_func(string)
 {
 enr_type, type  string;
 If type = "unsignedbyte" then enr_type = "byte";
 If type = "byte" then enr_type = "sbyte";
 If type = "unsignedint" then enr_type = "uint";
 If type = "unsignedshort" then enr_type = "ushort";
 If type = "unsignedlong" then enr_type =  "ulong";
}
```

Fig. 6. Enrichment Datatype Function

As result for the propped algorithm we get a new WSDL schema called (Enrichment Schema), shown in Figure 7.

```
enr_schema OperationName(ParName, type, enr_type){
<operation OperationName = "n1">
   <input ParName= "n2">
    <type>  type  </type>
    <enr_type> enr_type </enr_type>
    <min_enr_appearance> min </min_enr_appearance>
    <max_enr_appearance> max </max_enr_appearance>
   </input>
   <output OperationNameResponse = "n3">
    <type> type </type>
    <enr_type> enr_type </enr_type>
   </output>
</operation>  }
```

Fig. 7. Enrichment Schema

2) Visual Implementation for The Proposed Tool (WSDL_ET)

In this section we present our proposed tool as visual interfaces screens. We operated the tool as a case implementation for the three datatypes classifications; the output for our tool is a new enrichment XML_Schema with more simplification.The parameters datatypes is classified in three groups:

a) Clear Datatypes : The parameters datatypes are clear and need no modifications. The datatypes appear in the proposed enrichment schema as in the original WSDL document but with more simplification. The parameters datatypes are implemented inside WSDL as:

a. <s:element minOccurs="0" maxOccurs="1" name=" **stringParam**" type="**s:string**"/>
b. <s:element minOccurs="0" maxOccurs="1" name=" ClearWebServiceResult" type="**s:string**"/>

the element *StringParam* has a clear *string* type represented in a formal way, so, no modifications are needed. The enrichment schema for this *StringParam* is show in Figure 8 but with more simplifications.

```
enr_schema_ClearWebService ("stringParam", string, string)
{
  <operation OperationName="ClearWebService">
   <input ParName = "stringParam">
    <type>string</type>
    <enr_type>string</enr_type>
            <min_enr_appearance> 0
</min_enr_appearance>
            <max_enr_appearance>
1</max_enr_appearance>
 </input>
 <output OperationNameResponse=
"ClearWebServiceResponse">
   <type>string</type>
   <enr_type>string</enr_type>
 </output>
</operation>
  }
```

Fig. 8. Enrichment schema for (*ClearWebService)*

The enrichment schema in Figure 8 consists of three parts, the operation name and its parameters names and also each parameters datatypes with its constraints. These parts help the requester to know what does the Web Service serve, and also what are the parameters datatypes needed.

b) Indistinguishable Datatypes:

In this case of datatypes, the parameters datatypes is implemented using a custom datatypes *(ns : where ns is .NET namespace)* which is not a formal representation for the datatypes. WSDL document for (*IndistinguishableWebService)* is implemented inside WSDL as

a. <s:element minOccurs="0" maxOccurs="1" name="**byteParam**" type="**s:unsignedByte**"/>
b. <s:element minOccurs="0" maxOccurs="1" name="IndistinguishableWebServiceResult" type="s: unsignedByte "/>

the element *ByteParam* have a *unsignedbyte* type which is represented using .NET namespace and that representation is not a formal way. Additionally the Web Service provider has not selected these types, so that these parameters datatypes are needed to represent in a formal and simple representation. The proposed tool is run and converts all these .NET namespace datatypes and presents them in enrichment schema. The enrichment schema for this *ByteParam* is shown in Figure 9 with more specification and more simplifications.

```
enr_schema_IndistinguishableWebService (“byteParam”,
unsignedByte, Byte){
<operation
OperationName="IndistinguishableWebService">
  <input> ParName = "byteParam">
  <type>unsignedByte</type>
  <enr_type>Byte</enr_type>
            <min_enr_appearance>1</min_enr_appearance>
            <max_enr_appearance> 1
</max_enr_appearance>  </input>
  <output OperationNameResponse=
"IndistinguishableWebServiceResponse">
    <type>unsignedByte</type>
     <enr_type>Byte</enr_type>  </output>
 </operation> }
```

Fig. 9. Enrichment schema for (*IndistinguishableWebService)*

c) Unclear Datatypes:

This is the last classification of parameters datatypes. In this case the tool sends an interface application to Web Service provider to determine the selected datatypes, because the tool cannot guess which datatypes the provider selected for the specific parameters. WSDL document for (IndistinguishableWebService) is implemented inside WSDL as:

a. <s:sequence><s:element minOccurs="0" maxOccurs="1" name="IntList" type="tns:ArrayOfInt"/>
b. <s:element minOccurs="0" maxOccurs="unbounded" name="int" type="s:int"/>
c. <s:element name=" UnclearWebServiceResponse">
<s:element minOccurs="0" maxOccurs="1" name="UnclearWebServiceResult" type=" tns:ArrayOfInt "/>

The element *IntList* has a *ArrayOfInt* type which is represented using .NET namespace and that representation is not a formal way and is ambiguous, These parameters datatypes need to be presented in a formal and clear representation. The proposed tool is run and converts all these .NET namespace datatypes by returning back to the Web Service provider and asking to select from a list the datatypes for specific parameters. Figure 10 shows the interface which the tool uses to ask the Web Service provider to select the specific datatype.

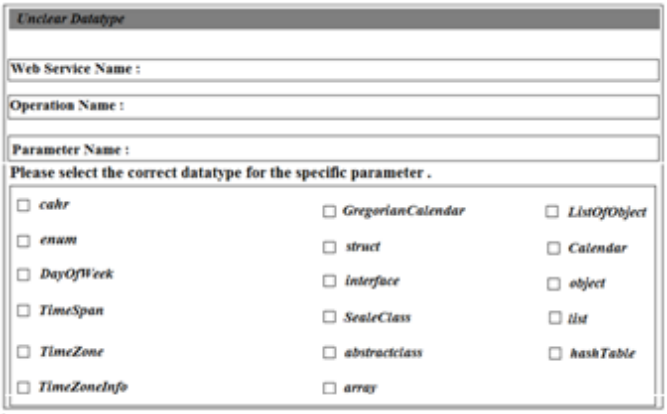


Fig. 10. Provided Datatype Interface

When the provider selects the specific datatype, the tool starts to create a new enrichment WSDL document with clearer parameters datatypes. The enrichment schema for the WSDL document which appeared in Figure 8 is shown in Figure 11 after the tool completed its functionality

```
enr_schema_UnclearWebService (“IntList”, ArrayOfInt, List)
{
 <operation
OperationName="IndistinguishableWebService">
  <input ParName = "IntList“>
   <type>ArrayOfInt</type>
   <enr_type>List</enr_type>
            <min_enr_appearance>0</min_enr_appearance>
            <max_enr_appearance>
1</max_enr_appearance>
  </input>
  <output OperationNameResponse=
"UnclearWebServiceResponse">
  <type> ArrayOfInt </type>
  <enr_type> List </enr_type>
 </output>
</operation>
}
```

Fig. 11. Enrichment schema for *(UnclearWebService)*

V. Conclusion and Future Work

The output of our paper is a tool and its algorithm for extending the XML-Schema to represent the operations parameters to reach better comprehension for the Web Service

functionality. Unlike previous approaches, which give a semantic for the Web Services during its WSDL documents, ignoring necessary of the operations parameters datatypes declarations, our approach can analyze all the operations of the Web Service and then classify all input output parameters which operations need. The tool deals with 33 datatypes and breaks them into three categories of datatypes discussed in chapter 4.

A. Conclusions

One of the problem that still facing Web Service is that the datatypes specification is difficult to understood and reuse by service requester. This paper had proposed an approach to solve this problem, the approach is based on the following:

1) Analyzing the XSD based datatypes inside WSDL produced by the .NET platform for different prototype Web Service .

2) Classify the datatypes specification into the following Categories:

a) Clear Datatypes : This Category includes the datatypes can easily be understood by service requester.

Table 6 shows these datatypes and how they implemented inside WSDL.

TABLE VI. CLEAR DATATYPES

Datatype	Implemented inside WSDL
Int	<xs:element minOccurs="0" name="intpar" type="xs:int" />
Short	<xs:element minOccurs="0" name="shortpar" type="xs: short"/>
Long	<xs:element minOccurs="0" name="longPar" type="xs: long"/>
Double	<xs:element minOccurs="0" name="doublePar" type="xs:double"/>
Boolean	<xs:element minOccurs="0" name="boolPar" type="xs: boolean"/>
Decimal	<xs:element minOccurs="0" name="decimalPar" type="xs:decimal"/>
String	<xs:element minOccurs="0" name="stringPar" nillable="true"type="xs:string"/>
timeZone	<xs:element minOccurs="0" name="arg0" type="tns: timeZone"/>
Float	<xs:element minOccurs="0" name="floatPar" type="xs: float"/>

a) Indistinguishable Datatypes : This category include the datatypes that can be distinguished by the proposed tool, Table 7 shows these datatypes and how they implemented inside WSDL.

TABLE VII. INDISTINGUISHABLE DATATYPES

Datatype	Implemented inside WSDL
Byte	<xs:element minOccurs="0" " name="bytepar" type="**xs:unsignedByte**" />
Sbyte	<xs:element minOccurs="0" name="sbytepar" type="**xs:byte**"/>
Uint	<xs:element minOccurs="0" name="uintPar" type="**xs:unsignedInt**"/>
Ushort	<xs:element minOccurs="0" name="ushortPar" type="**xs:unsignedShort**"/>
Ulong	<xs:element minOccurs="0" name="ulongPar" type="**xs:unsignedLong**"/>

b) Unclear Datatypes : This category include the datatypes that is difficult to be understood by the requester and also cannot be distinguished by the proposed approach . Table 8 shows these datatypes and how they implemented inside WSDL.

TABLE VIII. UNCLEAR DATATYPES

Datatype	Implemented inside WSDL
Char	<xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/" minOccurs="0" name="charInput" type="**q1:char**"/>
Object	<xs:element minOccurs="0" name="objectPar" nillable="true" type="**xs:anyType**"/>
Enum	<xs:element xmlns:q5="http://schemas.datacontract.org/2004/07/TeamProject1" minOccurs="0" name="enumPar"type="**q5:DayofWeek**"/>
DateTime	<xs:element minOccurs="0" name="datePar" type="**xs:dateTime**"/>
DayOfWeek	<xs:element xmlns:q7="http://schemas.datacontract.org/2004/07/System" minOccurs="0" name="dayPar"type="**q7:DayOfWeek**"/>
TimeSpan	<xs:element xmlns:q9="http://schemas.microsoft.com/2003/10/Serialization/" minOccurs="0" name="timeSpanPar"type="**q9:duration**"/>
Calendar	<xs:element name="arg0" type="**xs:dateTime**" minOccurs="0"/>
TimeZone	<xs:element xmlns:q11="http://schemas.datacontract.org/2004/07/System" minOccurs="0" name="timeZonePar"nillable="true" type="**q11:TimeZone**"/>
GregorianCalendar	<xs:element name="arg0" type="**xs:dateTime**" minOccurs="0"/>
TimeZoneInfo	<xs:element xmlns:q13="http://schemas.datacontract.org/2004/07/System" minOccurs="0" name="timeZoneInforPar"nillable="true" type="**q13:TimeZoneInfo**"/>

Struct	<xs:element xmlns:q16="http://schemas.datacontract.org/2004/07/TeamProject1" minOccurs="0" name="structPar"type="**q16:ValType1**"/>
Interface	<xs:element minOccurs="0" name="interfcaePar" nillable="true"type="**xs:anyType**"/>
Sealed Class	<xs:element xmlns:q1="http://schemas.datacontract.org/2004/07/TeamProject1" minOccurs="0" name="sealedPar"nillable="true" type="**q1:SealedClass**"/>
Abstract Class	<xs:element xmlns:q2="http://schemas.datacontract.org/2004/07/TeamProject1" minOccurs="0" name="abstractPar"nillable="true" type="**q2:AbstractClass**"/>
Arrays	<xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays" minOccurs="0" name="arrayPar" nillable="true" type="**q3:ArrayOfint**"/>
List	<xs:element xmlns:q5="http://schemas.microsoft.com/2003/10/Serialization/Arrays" minOccurs="0" name="listPar" nillable="true" type="**q5:ArrayOfstring**"/>
Hash Table	<xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays"minOccurs="0" name="hashPar" nillable="true"type="**q1:ArrayOfKeyValueOfanyTypeanyType**"/>
List of Object	<xs:element xmlns:q23="http://schemas.datacontract.org/2004/07/TeamProject1"minOccurs="0" name="getTeamsResult" nillable="true" type="**q23:ArrayOfTeam**"/>

3) Enriching the datatypes specification by producing an XML document of the enrichment specification depending on the following :

a) For the clear datatypes it will be remain the same in the result XML document.

b) In the case be indistinguishable datatypes the approach will use rules or conditions to decide the enrichment datatype as explained in the examples of chapter 5 .

c) For the unclear datatypes, in this case the provider is asked to select the prepare the enrichment to be put in the resulted XML. So the provider in intervention is limited to this category of unclear specification.

Based on this paper approach, a proof of concept tool had been built that can use any WSDL document as input and then produced the enriched datatype specification based on it.

B. Future work

Future work will concentrate on the following:

1) This research had depended merely on C# datatypes, however the future work will consider other languages datatypes such as Java or VB., etc.

2) Comparing the datatypes specification inside WSDL documents when using different programming languages to produce Web Services.

3) Enhancing the tool to enable it to work with datatypes specification produced by different programming languages.

ACKNOWLEDGMENT

The authors thank all the professors in faculty of information technology in Philadelphia University, especially prof. Said Ghoul and Dr. Nameer Emam for their scientific and accurate help.

REFERENCES

[1] Houda EL Bouhissi, Mimoun Malki. Reverse Engineering Existing web Services Applications, 16 th Working conference on Reverse Engineering, 2009.

[2] Hongbing Wang, Joshua Zhexue Huang, Yuzhong Qu, Junyuan Xie. Web services: Problems and Future Directions, Elsevier, 2004.

[3] Jinghai Rao, Su Xiaomeng. A Survey of Automated Web Service Composition Methods, In Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition, SWSWPC 2004.

[4] Roberto De Virgilio. Meta-Modeling of Semantic Web Services, IEEE International Conference on Services Computing, DOI 10.1109/SCC.2010.22, 2010.

[5] Samer Hanna. Web services robustness testing, Ph.D, Durham theses, Durham University, (2008).

[6] Wei-Tek Tsai, Yinong Chen, Ray Paul. Specification-Based Verification and Validation of Web Services and Service-Oriented Operating Systems , 10th IEEE International Workshop on Object-oriented Real-time Dependable Systems (WORDS 05), Sedona, pp. 139 – 147, February 2005.

[7] Jia Zhang, Liang-Jie Zhang. Editorial Preface: Web Services Quality Testing, International Journal of Web Services Research, April-June 2005.

[8] Weider D. Yu, Aravind, D. & Supthaweesuk, P. Software Vulnarability Analysis for Web Services Software Systems. Proceedings of the 11th IEEE, 2006.

[9] Glenford Myers. The Art of Software Testing, ISBN 0-471-04328-1, John Wiley. Neumann, P. (2004). Principled assuredly trustworthy composable architecture.

[10] John Timm T. E., Gerald C. Gannod. Specifying Semantic Web Service Compositions using UML and OCL, IEEE International Conference on Web Services (ICWS), 2007.

[11] Eladio Domínguez, Jorge Lloret, Beatriz Pérez, Áurea Rodríguez, Ángel L. Rubio and María A. Zapata. A Survey of UML Models to XML Schemas Transformations, Lecture Notes in Computer Science Volume 4831, pp 184-195, 2007.

[12] Weijun Sun, Shixian Li Defen Zhang, YuQing Yan. A Model-driven Reverse Engineering Approach for Semantic Web Services Composition, World Congress on Software Engineering, DOI 10.1109/WCSE.2009.403, 2009.

[13] Alexandre Bellini, Antonio Francisco do Prado, Luciana Aparecida Martinez Zaina. Top-Down Approach for Web Services Development, Fifth International Conference on Internet and Web Applications and Services, 2010.

[14] Evren Sirin, Bijan Parsia and James Hendler. Composition-driven Filtering and Selection of Semantic Web Services, American Association for Artificial Intelligence, 2004.

[15] Roy Grønmo, David Skogan, Ida Solheim, Jon Oldevik. Model-driven Web Service Development, International Journal of Web Services Research, 1(4), Oct-Dec 2004.

[16] Juan Vara, Valeria De Castro and Esperanza Marcos. WSDL Automatic Generation from UML Models in a MDA Framework, International Journal of Web Services Practices, Vol.1, No.1-2 (2005), pp. 1-12.

[17] Samer Hanna and Ali Alawneh. An Approach of Web Service Quality Attributes Specification, Communications of the IBIMA Journal (ISSN: 1943-7765), 2010.

Modifications of Particle Swarm Optimization Techniques and Its Application on Stock Market: A Survey

Razan A. Jamous
Pure Mathematics - Department of Mathematics, Faculty of Science, Ain Shams University
Cairo, Egypt

Assem A. Tharwat
Departments of Operations Research and Decision Support, Faculty of Computer and Information,
Cairo University

EssamEl.Seidy
Pure Mathematics - Department of Mathematics, Faculty of Science, Ain Shams University
Cairo, Egypt

Bayoumi Ibrahim Bayoum
Pure Mathematics - Department of Mathematics, Faculty of Science, Ain Shams University
Cairo, Egypt

***Abstract*—Particle Swarm Optimization (PSO) has become popular choice for solving complex and intricate problems which are otherwise difficult to solve by traditional methods. The usage of the Particle Swarm Optimization technique in coping with Portfolio Selection problems is the most important applications of PSO to predict the stocks that have maximum profit with minimum risk, using some common indicators that give advice of buy and sell. This paper gives the reader the state of the art of the various modifications of the PSO and study whether had been applied over the stock market or not.**

Keywords—Computational intelligence; Particle Swarm Optimization; modification; Stock Market; Portfolio Selection

I. Introduction

Computational Intelligence (CI) is the study of adaptive mechanisms to enable or facilitate intelligent behavior in complex and changing environments. Studies of social animals and social insects have resulted in a number of computational models of swarm intelligence. Biological Swarm Systems that have inspired computational models include ants, bees, spiders, and bird flocks [1] The authors in]. The objective of computational swarm intelligence models is to modeling the simple behaviors of individuals, and the local interactions with the environment and neighboring individuals, in order to obtain more variant behaviors that can be used to solve complex problems, mostly optimization problems. Swarm intelligence (SI) originated from the study of colonies, or swarms of social organisms. Studies of the social behavior of organisms (individuals) in swarms prompted the design of very efficient optimization and clustering algorithms. For example, simulation studies of the graceful, but unpredictable, choreography of bird flocks led to the design of the Particle Swarm Optimization (PSO) algorithm [2]. However, in that short period, PSO has gained widespread appeal amongst researchers and has been shown to offer good performance in a variety of application domains. The usage of PSO in stock market and Portfolio Selection is very common today, whereas, the average person's interest in the stock market has grown exponentially. This demand coupled with advances in trading technology has opened up the markets, so that nowadays anybody can own stocks, and use many types of software to perform the aspired profit with minimum risk. Consequently, a lot of attention has been devoted to the analysis and prediction of future values and trends of the financial markets, and due to large applications in different business transactions, stock market prediction has become a hot topic of research. Nowadays, more software such as PSO are used to guide person to manage his portfolio and get successful investment, this motivates researchers to develop these software to give more accuracy and efficiency for successful portfolio management. In this paper we survey the state of the art of the various modifications of the PSO and study whether had been applied over the stock market or not. The rest of the paper is organized as follows. Section 2 gives historical study of particle swarm optimization. Section 3 briefly reviews the stock market. In section 4, the basic particle swarm optimization is presented. Section 5 gives the Variations of Particle Swarm Optimization and explains how it can modified PSO and the terms which we can modify it. Also, it gives the different forms for the Modifications of the original PSO. Finally, in Section 6 we conclusion this paper by the summary of main points.

II. Historical Background of Particle Swarm Optimization

Kennedy and Eberhart introduced particle Swarm Optimization (PSO) in 1995 as a stochastic optimization algorithm based on social simulation model [3]. Since its inception in 1995, research and application interest in PSO have increased, resulting in an exponential increase in the number of publications, Parsopoulos and Vrahatis provided statistical study about the exponential increase in number of publications about PSO during the year 2000 to 2013, in this work the statistical study was completed for the next two years, and the increase in number of PSO publications is still exponential as shown in Figure1.

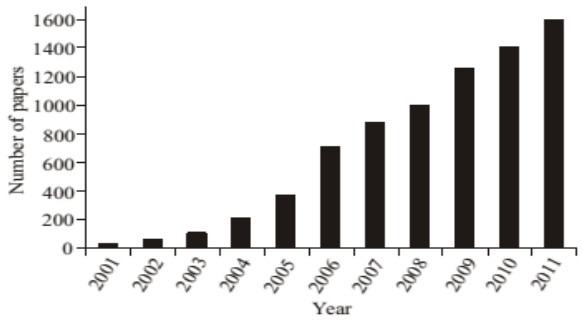


Fig. 1. Number of journal papers with the term "particle swarm" in their titles, published by three major publishers, namely Elsevier, Springer, and IEEE, during the years 2000-2011

Research in PSO has resulted in a large number of new PSO algorithms that improves the performance of the original PSO and enables application of PSO to different optimization problem types (e.g., unconstrained optimization, optimization in dynamic environments, constrained optimization, multi-objective optimization and finding multiple solutions). Elaborate theoretical studies of PSO dynamics have been done, and PSO parameter sensitivity analyses have resulted in a better understanding of the influence of PSO control parameters. PSO applications vary in complexity and cover a wide range of application areas. The PSO algorithm simulates the behaviors of bird flocking, the flight of a bird flock can be simulated with relative accuracy by simply maintaining a target distance between each bird and its immediate neighbors. This distance may depend on its size and desirable behavior. PSO learned from this and used it to solve the optimization problems. Therefore in PSO, each single solution (a bird in the search space) is called a particle, and each particle has fitness value which is evaluated by the objective function to be optimized, and has a velocity which directs the flying of the particle. All particles fly through the problem space by following the current optimum particle.

III. Stock Market Review

Stock market is, without a doubt, one of the greatest tools ever invented for building wealth. Stocks are a part, if not the cornerstone, of any investment portfolio[4]. This demand coupled with advances in trading technology has opened up the markets so that nowadays nearly anybody can own stocks, and use many types of software to perform the aspired profit with minimum risk. Consequently, a lot of attention has been devoted to the analysis and prediction of future values and trends of the financial stock markets, and due to large applications in different business transactions, stock market prediction has become a hot topic of research.

IV. Prediction of the Stock Market

A. Defining the prediction task

Before having any further discussion about the prediction of the market we define the task in a formal way. Following [15] Given a sample of N examples $\{(x_i, y_i), i=1,\ldots,N\}$ where $f(x_i)= y_i$, $\forall i$,return a function g that approximates f in the sense that the norm of the error vector $E=(e_1,\ldots,e_N)$ is minimized. Each e_i is defined as $e_i=e(f(x_i), y_i)$ where e is an arbitrary error function". In other words the definition above indicates that in order to predict the market you should search historic data and find relationships between these data and the value of the market. Then try to exploit these relationships you have found on future situations. This definition is based on the assumption that such relationships do exist.

B. Prediction Methods

The prediction of the market is without doubt an interesting task. In the literature there are a number of methods applied to accomplish this task. These methods use various approaches, ranging from highly informal ways (the study of a chart with the fluctuation of the market) to more formal ways (linear or non-linear regressions).We have categorized these techniques as follows:

- Technical Analysis Methods.
- Fundamental Analysis Methods.
- Traditional Time Series of Prediction Methods.
- Machine Learning Methods.

The criterion to this categorization is the type of tools and the type of data that each method is using in order to predict the market.

C. Technical Analysis

Following [5] "Technical analysis is the method of predicting the appropriate time to buy or sell a stock used by those believing in the castles-in-the-air view of stock pricing". The idea behind technical analysis is that share prices move in trends dictated by the constantly changing attributes of investors in response to different forces. Using technical data such as price, volume, highest and lowest prices per trading period the technical analyst uses charts to predict future stock movements. Price charts are used to detect trends, and these trends are assumed to be based on supply and demand issues which often have cyclical or noticeable patterns. From the study of these charts trading rules are extracted and used in the market environment. The technical analysts are known as "chartists". Most chartists believe that the market is only 10 percent logical and 90 percent psychological [5].

D. Fundamental Analysis

Following [5] "Fundamental analysis is the technique of applying the tenets of the firm foundation theory to the selection of individual stocks". The analysts that use this method of prediction use fundamental data in order to have a clear picture of the firm (industry or market) they will choose to invest on. They are aiming to compute the "real" value of the asset that they will invest in and they determine this value by studying variables such as the growth, the dividend payout, the interest rates, the risk of investment, the sales level, tax rates and so on. Their objective is to calculate the intrinsic value of an asset (e.g. of a stock). Since they do so they apply a simple trading rule. "If the intrinsic value of the asset is higher than the value it holds in the market, invest in it. If not, consider it a bad investment and avoid it". The fundamental analysts believe that the market is defined 90 percent by logical and 10 percent by physiological factors. This type of analysis is not possible to fit in the objectives of our study. The reason for this is that the data it uses in order to determine the intrinsic

value of an asset does not change on daily basis. Therefore fundamental analysis is helpful for predicting the market only in a long-term basis.

E. Traditional Time Series Prediction

The Traditional Time Series Prediction analyzes historical data and attempts to approximate future values of a time series as a linear combination of these historical data .In econometrics there are two basic types of time series forecasting: univariate (simple regression) and multivariate (multivariate regression) [6].These types of regression models are the most common tools used in econometrics to predict time series. The way they are applied in practice is that firstly a set of factors that influence (or more specific is assumed that influence) the series under prediction is formed. These factors are the explanatory variables xi of the prediction model. Then a mapping between their values x_{it} and the values of the time series y_t(y is the to-be explained variable) is done, so that pairs $\{x_{it}, y_t\}$ are formed. These pairs are used to define the importance of each explanatory variable in the formulation of the to-be explained variable. In other words the linear combination of xi that approximates in an optimum way y is defined. Univariate models are based on one explanatory variable (I=1) while multivariate models use more than one variable (I>1).

To sum up, it is possible to apply this methodology to predict the market on a daily basis. Additionally it is widely used by the economists and therefore it is a methodology that we can use for the purposes of the present study.

F. Machine Learning Methods

Several methods for inductive learning have been developed under the common label "Machine Learning". All these methods use a set of samples to generate an approximation of the underling function that generated the data. The aim is to draw conclusions from these samples in such way that when unseen data are presented to a model it is possible to infer the to-be explained variable from these data. From these methods here are: The Nearest Neighbor and the Neural Networks Techniques. Both of these methods have been applied to market prediction; particularly for Neural Networks there is a rich literature related to the forecast of the market on daily basis [7].

V. Prediction Techniques

Many research papers have appeared in the literature using evolutionary computing tools such as genetic algorithm (GA), particle swarm optimization (PSO), bacterial foraging optimization (BFO) and genetic programming (GP) in developing forecasting models. Hassan et al. described a novel time series forecasting tool, their fusion model combines a Hidden Markov Model (HMM), Artificial Neural Networks (ANN) and Genetic Algorithms (GA) to forecast financial market behavior [12]. In another work, Aboueldahab et al. introduced a new Enhanced Particle Swarm Optimization (EPSO) to train the Sigmoid Diagonal Recurrent Neural Networks (SDRNN) weights and applied this technique in the forecasting of both NASDAQ100 and S&P500 stock market indices [9]. Majhiet al. used the standard particle swarm optimization (PSO) algorithm to develop an efficient forecasting model for prediction of S&P500 and DJIA stock indices [13]. The connecting weights of the adaptive linear combiner based model are adjusted by the PSO so that its mean square error (MSE) is minimized. In another publication [13] Majhi developed two new forecasting models based on bacterial foraging optimization (BFO) and adaptive bacterial foraging optimization (ABFO) to predict S&P500 and DJIA stock indices using technical indicators derived from the past stock indices. The structure of these models is basically an adaptive liner combiner, the weights of trained using the ABFO and BFO algorithms.

VI. Basic Particle Swarm Optimization

Individuals in a particle swarm follow a very simple behavior: to emulate the success of neighboring individuals and their own successes. The collective behavior that emerges from this simple behavior is that of discovering optimal regions of a high dimension al search space. PSO algorithm maintains a swarm of particles, where each particle represents a potential solution. In analogy with evolutionary computation paradigms, a swarm is similar to a population, while a particle is similar to an individual. In simple terms, the particles are "flown" through a multidimensional search space, where the position of each particle is adjusted according to its own experience and that of its neighbors.

The following description of the PSO algorithm is adapted from [14]. Let $X_i(t)= (x_{i1}, x_{i2}, \ldots, x_{id})$ denote the position of particle i in the search space at time step t, $V_i(t) = (v_{i1}, v_{i2}, \ldots, v_{id})$ denote the velocity particle i in the search space at time step t, $P_i= (p_{i1}, p_{i2}, \ldots, p_{id})$ denote the best solution achieved so far by the particle itself, $P_g= (p_{g1}, p_{g2}, \ldots, p_{gd})$denote the best solution achieved so far by the whole swarm. The new position of the particle is changed by adding a velocity to the current position, as follows:

$$x_{id}^{(t+1)} = x_{id}^{(t)} + v_{id}^{(t+1)} \quad (1)$$

$$v_{id}^{(t+1)} = w.V_{id}^{(t)} + c_1 r_1\left(P_{id} - X_{id}^{(t)}\right) + c_2 r_2\left(P_{gd} - X_{id}^{(t)}\right) \quad (2)$$

Where c_1 and c_2 are two positive constants, r_1 and r_2 are two random numbers in the range [0, 1]; w is the inertia weight. The velocity vector drives the optimization process, and reflects both the experiential knowledge of the particle and socially exchanged information from the particle's neighborhood. The experiential knowledge of a particle is generally referred to as the cognitive component, which is proportional to the distance of the particle from its own best position (referred to as $pbest_i$). The socially exchanged information is referred to as the social component of the velocity equation (2), which is proportional to the distance of the particle from the best position found by the swarm (referred to as gbest).

A. Global Best PSO

For the global best PSO, or gbest PSO, the neighborhood for each particle is the entire swarm. The social component of the particle velocity update reflects information obtained from all the particles in the swarm. In this case, the social information is the best position found by the swarm.

B. Local Best PSO

For the local best PSO, or $pbest_i$ PSO, the neighborhood for each particle is small number of particles in the swarm. So that the social component reflects information exchanged within the neighborhood of the particle, reflecting local knowledge of the environment. In this case, the social information is the best position found by the experiential knowledge of the particle.

C. Velocity Components

The velocity calculation as given in equation (2) consists of three terms:

- The previous velocity, $V_i(t)$, which serves as a memory of the previous flight direction, i.e. movement in the immediate past. This memory term can be seen as a momentum, which prevents the particle from changing direction, and to bias towards the current direction. This component is also referred to as the inertia component.
- The cognitive component, c1r1(Pi–Xi), which quantifies the performance of particle irrelative to past performances. In a sense, the cognitive component resembles individual memory of the position that was best for the particle. The effect of this term
- is that particles are drawn back to their own best positions, resembling the tendency of individuals to return to situations or places that satisfied them most in the past.
- The social component, c2r2(Pg–Xi), which quantifies the performance of particle I relative to a group of particles, or the swarm. Conceptually, the social component resembles a group norm or standard that individuals seek to attain. The effect of the social component is that each particle is also drawn towards the best position found by whole the swarm.

VII. Modifications of the Original PSO

We divide different modifications on PSO into two main categories, external modifications and internal modifications.

A. External Modification Techniques

External Modification interests with all modifications which perform not on the basic components of PSO such as the method which use multi swarms or methods which split the swarm.

1) Dynamic multi-swarm particle swarm optimizer

The authors in [15] have proposed DMS-PSO based on the new neighborhood topology. In this method the whole of the population are divided into small sized swarm. Each sub-swarm uses its own members to search for better regions in the search space. In order to increase the diversity these sub-swarms are regrouped frequently to exchange the information among all particles. A local search is combined with the algorithm to improve the overall algorithm's local searching ability. The DMS-LPSO is tested on a set of benchmark functions and the results show that the proposed algorithm can find reasonable solutions for all of the problems.

2) Multi-swarm and multi-best particle swarm optimization algorithm

A new method named Multi-Best PSO (MBPSO) is proposed [16]. This method instead of using single global best position (gbest) and personal best position (Pbest), it uses the multi gbest and multi Pbest. So in the course of searching, other best values can help the best value trapped by local optimum fly out of local position. MBPSO divided the whole population into the sub-swarms and then calculates the several gbest and then combines all particles together and then calculates again taking the result as a new initial value.

3) Dynamic multi-swarm particle swarm optimizerwith sub-regional harmony search

DMS-PSO-SHS [33] is an extension of DMS-PSO that divided the whole of the population into the small sub-swarm with dynamic size to adopt each one the population of the harmony search algorithm. This method based on the DMS-PSO, generate new harmonies according to the current personal best solution and the nearer personal best solution is replaced with a new harmony with better fitness. The DMS-PSO-SHS enables the particles to have more diverse exemplars to learn from after we frequently regroup the swarms and allow the harmonies to search in a larger potential space among different sub-populations.

4) Multi-swarm Particle Swarm Optimization

In [16], the authors have proposed a Multi-swarm Particle Swarm Optimization (MPSO) to maintain the swarm diversity. This method applied a mixed local search behavior modes and information exchange among subswarms. When the premature convergence occurs in one sub-swarm then that particles should escape from the local area through the initialization their position in the search space.

5) Master-slave swarm evolutionary (MSSE-PSO)

The authors in [17] developed a shuffling master-slave swarm evolutionary algorithm based on particle swarm optimization (MSSE-PSO). The population is sampled randomly from the feasible space and partitioned into several sub-swarms (one masters warm and additional slave swarms), in which each slaves warm independently executes PSO. The master swarm is enhanced by the social knowledge of the master swarm itself and that of the slave swarms.

6) Heterogeneous partical swarm Optimization (HPSO)

The authors in [13] introduced a Heterogeneous PSO (HPSO). In the standard PSO and most of its modifications, particles follow the same behaviors. That is, particles implement the same velocity and position update rules and they exhibit the same search characteristics. In HPSO particles are allowed to follow different search behaviors in terms of the particle position and velocity updates selected from a behavior pool, thereby efficiently addressing the exploration-exploitation trade off problem. Two versions of the HPSO were proposed, the one static, where behaviors do not change, and a dynamic version where a particle may change its behavior during the search process if it cannot improve its personal best position.

7) BP algorithm

In optimizing the particle swarm optimization (PSO) that inevitable existence problem of Prematurity and the local convergence. Based on these aspects, [18] proposed a kind of modified particle swarm optimization algorithm, they take the gradient descent method (BP algorithm) as a particle swarm operator embedded in particle swarm algorithm, and at the same time they use to attenuation wall (Damping) approach to make fly off the search area of the particles of size remain unchanged and avoid the local optimal solution, with three input XOR problem to testing the improvement of the particle swarm.

B. Internal Modification Techniques

Internal Modification interests with the modifications which happened on the basic components of PSO.A number of basic modifications to the basic PSO have been developed to improve speed of convergence and the quality of solutions found by the PSO. These modifications include the introduction of an inertia weight, velocity clamping, velocity constriction, different ways of determining the global best and the local best positions, and different velocity models.

1) Velocity Clamping

One of the important aspects that determines the efficiency and accuracy of an optimization algorithm is the exploration–exploitation trade-off. Exploration is the ability of a search algorithm to explore different regions of the search space in order to locate a good optimum. Exploitation, on the other hand, is the ability to concentrate the search around a promising area in order to refine a candidate solution. A good optimization algorithm optimally balances these contradictory objectives. Within the PSO, these objectives are addressed by the velocity update equation. The velocity updates in equations (2) consist of three terms that contribute to the step size of particles. In the early applications of the basic PSO, it was found that the velocity quickly explodes to large values, especially for particles far from the local best and global best positions. Consequently, particles have large position updates, which result in particles leaving the boundaries of the search space (the particles diverge). To control the global exploration of particles, velocities are clamped to stay within boundary constraints [19]. If a particle's velocity exceeds a specified maximum velocity Vmax, the particle's velocity is set to the maximum velocity. However, the problem of finding a good value for each Vmaxin order to balance between moving too fast or too slow, and exploration/exploitation. Usually, the Vmax values are selected to be a fraction of the domain of each dimension of the search space Xmax and Xmin, and calculated as follows:

$$\mathrm{Vmax} = \delta(\mathrm{Xmax} - \mathrm{Xmin}) \quad (3)$$

Where Xmax and Xmin are respectively the maximum and minimum value of the domain, and $\delta \in [0, 1]$. The value of δ is problem dependent, as was found in a number of empirical studies of Shi and Eberhart [20].

a) Dissipative Particle Swarm Optimization

In order to prevent premature convergence [21] proposed Dissipative PSO (DPSO) by adding random mutation to PSO. This could be thought of as an inspiration for GA. DPSO introduces negative entropy through the addition of randomness to the particles. The results showed that DPSO performed better than standard PSO when applied to the benchmark problems.

b) Particle Swarm Optimization with passive congregation

The authors in [22] Presented a PSO with passive congregation (PSOPC) to improve the performance of Standard PSO (SPSO). Passive congregation is an important biological force preserving swarm integrity. By introducing passive congregation to PSO, information can be transferred among individuals of the swarm. This approach was tested with a benchmark test and compared with standard Gbest mode PSO, Lbest mode PSO and PSO with a constriction factor, respectively. Experimental results indicate that the PSO with passive congregation improves the search performance on the benchmark functions significantly.

c) Stochastic Particle Swarm Optimization

A new particle swarm optimizer, called Stochastic PSO (SPSO), which is guaranteed to convergence to the global optimization solution with probability one, is presented based on the analysis of the standard PSO [23]. In this approach, if the global best position is replaced by a particle's position in some interaction, this particles' position will be regenerated and if a particles' new position coincides with the global best position, its position will also be regenerated randomly. The authors have proved that this is a guaranteed global convergence optimizer and through some numerical tests this optimizer show edits good performance.

d) Cooperative Particle Swarm Optimization

A modified particle swarm optimizer named Cooperative PSO (CPSO) was proposed by Van denBergh and Engelbrecht [24]. The CPSO could significantly improve the performance of the original PSO by utilizing multiple swarms for optimizing different components of the solution vector by employ in cooperative behavior. In this method, the search space is partitioned by dividing the solution vectors into small vectors, based on the partition several swarms will be randomly generated in different parts of the search space and used to optimize different parts of the solution vector.

e) Particle Swarm Optimization with disturbance term

[25] presented PSO with disturbance term (PSO-DT) which add a disturbance term to the velocity updating equation based on the prototype of the standard PSO trying to improve (or avoid) the shortcoming of standard PSO. The addition of the disturbance term based on existing structure effectively mends the defects. The convergence of the improved algorithm was analyzed. Simulation results demonstrated that the improved algorithm have a better performance than the standard one.

f) Center Particle Swarm Optimization

It was presented by Liu and his colleagues based on introducing a center particle to the LDWPSO algorithm [25]. The center particle is proposed explicitly to visit the center of the swarm at every iteration. After N-1particles update their positions as the usual PSO algorithms at every iteration, a center particle is updated according the following formula:

$$X_{cd}^{(t+1)} = \frac{1}{N-1}\sum_{i=1}^{N-1} X_{id}^{(t+1)} \quad (3)$$

Unlike other particles, the center particle has no velocity, but it is involved in all operations the same as the ordinary particle, such as fitness evaluation, competition for the best particle, except for the velocity calculation. The center particle has opportunities to become the gbest of the swarm. Hence it can guide the whole swarm to promising region and accelerate convergence. The center particle and a randomly selected ordinary particle were recorded during the optimization process. It was clear that the center particle has higher probability to be gbest. Therefore, the center particle often guides the search process, and although it is only one particle, it imposes great effect on the swarm.

g) Mean Particle Swarm Optimization

Deep and Bansal presented MeanPSO algorithm based on a novel philosophy of modifying the formula of velocity update equation [8]. The two terms of original velocity update formula were replaced by two new terms based on the linear combination $\left(\frac{P_i+P_g}{2}\right)$ and $\left(\frac{P_i-P_g}{2}\right)$ as follows:

$$V_i^{(t+1)} = V_i^{(t)} + c_1 r_1 \left(\left(\frac{P_i+P_g}{2}\right) - X_i^{(t)}\right) + c_2 r_2 \left(\left(\frac{P_i-P_g}{2}\right) - X_i^{(t)}\right) \quad (4)$$

Where the first term represents the current velocity of the particle (can be thought as a momentum term). The second term is proportional to the vector $\left(\left(\frac{P_i+P_g}{2}\right) - X_i^{(t)}\right)$ and is responsible for the attraction of particle's current position towards the mean of the positive direction of global best position (Pg) and positive direction of its own best position (Pi). The second term is proportional to the vector $\left(\left(\frac{P_i-P_g}{2}\right) - X_i^{(t)}\right)$ and responsible for the attraction of particle's current position towards the mean of the positive direction of its own best position (Pi) and the negative direction of the global best position (-Pg).

h) field-effect transistor (FET)

[26] introduces a modified particle swarm algorithm to handle multi objective optimization problems. In multi objective PSO algorithms, the determination of Pareto optimal solutions depends directly on the strategy of assigning a best local guide to each particle. In this work, the PSO algorithm is modified to assign a best local guide to each particle by using minimum angular distance information. This algorithm is implemented to determine field-effect transistor (FET) model elements subject to the Pareto domination between the scattering parameters and operation bandwidth. Furthermore, the results are compared with those obtained by the non-dominated sorting genetic algorithm-II. FET models are also built for the 3 points sampled from the different locations of the Pareto front, and a discussion is presented for the Pareto relation between the scattering parameter performances and the operation bandwidth for each model.

i) Fuzzy particle swarm optimization (FPSO)

Fuzzy particle swarm optimization (FPSO) [27] is a new variant of particle swarm optimization (PSO). Compared to PSO, each particle in FPSO is attracted by its previous best particle and other particles (not the global best particle) selected by a fuzzy mechanism. Although FPSO effectively slows down the attraction of the previous best particle and the global best particle, it shows slow convergence rate when solving complex optimization problems. To enhance the performance of FPSO, the authors propose an improved FPSO algorithm (IFPSO) which employs two strategies including generalized opposition-based learning (GOBL) and L´ evy mutation. In order to verify the performance of this approach, thirteen well-known benchmark functions and a real-world optimization problem are used in the experiments.

j) A Modified particle swarm optimization (MPSO)

The authors in [28] proposed a modified particle swarm optimization (MPSO) algorithm to solve the reliability redundancy optimization problems. The MPSO modifies the strategy of generating new position of particles. For each generation solution, the flight velocity of particles is removed. Whereas the new position of each particle is generated by using difference strategy. In addition, an adaptive parameter λ1 is used in MPSO. It can ensure diversity of feasible solutions to avoid premature convergence.

2) Inertia Weight

The inertia weight was introduced by Shi and Eberhart [11] as a mechanism to control the exploration and exploitation abilities of the swarm and as a mechanism to eliminate the need for velocity clamping. The inertia weight was successful in addressing the first objective, but could not completely eliminate the need for velocity clamping. The inertia weight, w, controls the momentum of the particle by weighing the contribution of the previous velocity, basically controlling how much memory of the previous flight direction will influence the new velocity. The value of w is extremely important to ensure convergent behavior, and to optimally tradeoff exploration and exploitation. For $w \geq 1$, velocities increase over time, accelerating towards the maximum velocity (assuming velocity clamping is used), and the swarm diverges. Particles fail to change direction in order to move back towards promising areas. For $w < 1$, particles decelerate until their velocities reach zero (depending on the values of the acceleration coefficients). Large values for w facilitate exploration, with increased diversity. A small w promotes local exploitation. However, too small values eliminate the exploration ability of the swarm. Little momentum is then preserved from the previous time step, which enables quick changes in direction. The smaller w, the more do the cognitive and social components control position updates.

a) Linear Decreasing Weight Particle Swarm Optimization

Linear Decreasing Weight particle swarm optimization (LDWPSO) algorithm was presented by Shi and Eberhart [23]. The inertia weight w is decreased linearly over the searching iterations, from an initial value to a final value as follows:

$$w = (w_{max} - w_{min}) \times \frac{(Max.Iter - Iter)}{Max.Iter} + w_{max} \quad (5)$$

Where w is the inertia weight that controls the velocity of particles, w_{max} is the initial inertia weight, w_{min} is the final inertia weight, Max. Iter is the maximum number of iterations,

and Iter is the current iteration. LDWPSO algorithm uses equation (1) to update position, equation (2) to update velocity and equation (5) to update the inertia weight.

b) Particle Swarm Optimization with Dynamic Adaptation

The author in [28] proposed another dynamic inertia weight to modify the velocity update formula in a method called modified Particle Swarm Optimization with Dynamic Adaptation (DAPSO).

c) Exponential Particle Swarm Optimization

The authors in [14], Ghali and his colleagues presented Exponential particle swarm optimization (EPSO) algorithm based on simple update in the form of inertia weight formula, as follows:

$$w = (w - 0.4) \times e^{-\frac{(Max.Iter - Iter)}{Max.Iter}} + 0.4 \qquad (6)$$

EPSO algorithm uses equation (1) to update position, equation (2) to update velocity and equation (6) to update the inertia weight.

d) C-Catfish PSO

Introduced chaotic maps into catfish particle swarm optimization. Swarm optimization (C-CatfishPSO) is a novel optimization algorithm proposed by [20].The [30] introduce a new parameter, called inertia weight, into the original particle swarm optimizer. Simulations have been done to illustrate the significant and effective impact of this new parameter on the particle swarm optimizer.

e) PSO with Nonlinear Decreasing inertia Weight (PSO-NDW)

Ultrasonic motor (USM) exhibits non-linearity that relates the input and output. It also causes serious characteristic changes during operation. PID controller has been widely used as the control scheme for USM. However, it is difficult for the fixed-gain type PID controller to compensate such characteristic changes and non-linearity of USM. [30] proposed a modified PSO with Nonlinear Decreasing inertia Weight (PSO-NDW) for optimal self-tuning of PID controller in positioning control of USM. A modified PSO employs the strategy that nonlinearly decreases the value of inertia weight from a large value to a small value. This strategy is to improve the performance of the standard PSO algorithm in global search and fine-tuning of the solutions. The performance of PSO-NDW based PID controller has been evaluated on the USM servo system. The results demonstrate that the proposed modified PSO can improve the accuracy of USM.

3) Acceleration Coefficients

A new approach was developed by Clerc and Kennedy, very similar to the inertia weight, to balance the exploration/exploitation trade-off, where the velocities are constricted by a constant χ, referred to as the constriction coefficient [31]. The velocity update equation changes to:

$$V_i^{(t+1)} = \chi \left[V_i^{(t)} + \varphi_1 \left(P_i - X_i^{(t)} \right) + \varphi_2 \left(P_g - X_i^{(t)} \right) \right] \qquad (7)$$

Where:

$$\chi = \frac{2k}{\left| 2 - \varphi - \sqrt{\varphi(\varphi - 4)} \right|} \qquad (8)$$

With $\varphi = \varphi_1 + \varphi_2, \varphi_1 = c_1 r_1$ and $\varphi_2 = c_2 r_2$. Equation (5) is used under the constraints that $\varphi \geq 4$ and $k \in [0, 1]$. The constriction approach was developed as a natural dynamic way to ensure convergence to a stable point, without the need for velocity clamping. Under the conditions that $\varphi \geq 4$ and $k \in [0, 1]$, the swarm is guaranteed to converge. The constriction coefficient χ evaluates to a value in the range [0, 1] which implies that the velocity is reduced at each time step.

a) Constrained Particle Swarm Optimization CPSO

In the other study, a Constrained Particle Swarm Optimization (CPSO) is developed by [32]. In this method, constraint handling is based on particle ranking and uniform distribution. For equality constraints, the coefficient weights are defined and applied for initializing and updating procedure. This method applied to schedule generation and reserve dispatch in a multi-area electricity market considering system constraints to ensure the security and reliability of the power system. CPSO applied to three case studies and results showed promising performance of the algorithm for smooth and non-smooth cost functions.

Table (I), shows a summary of modifications of the PSO and whether had been applied over the stock market or not.

THE DISCUSSION OF TABLE (I)

After having a careful look at the papers we reviewed, it is concluded that there has been notably a lot of work done and remains much more scopes and areas to work on the PSO and application aspects of PSO over stock market. So the implantation these modified methods over the stock market and study the performance of these methods and how it effect to support the decision making in the stock market is very critical issue.

Certain parameters require tuning to make PSO algorithm works well. However, changing PSO parameters can have a relatively large effect. Unadjusted particles' velocity may exceed a maximum value. Particles with such speeds might explore the search space, but lose the ability to fine-tune a result. The inertia weight or the random values also control the performance of the PSO algorithm. The higher the inertia weight, the higher the particle speed. As with the maximum velocity coefficient, the setting of the inertia weight must compromise between having a good exploration of the search space and a good fine-tuning ability.

We can see that the most modifications in the papers we reviewed done on the velocity of PSO which has the most important impact on the improvement of the PSO performance. So in our work we propose a new modification on the velocity equation of PSO to improve the convergence behavior of the Particle Swarm Optimization algorithm. Then based on our new modified PSO, we develop a new effective prediction model for stock markets and use this new model for solving portfolio optimization problem to provides a better safety investment in stock market and high prediction accuracy.

As it seen from table1, the work of some researchers were interested to make internal modification. But no one interested to apply external and internal modifications on PSO at the same time.

To improve PSO performance certain parameters have to be controlled. So, the most papers we reviewed interested to modify one of the basic components of the PSO i.e. velocity clamping, inertia term, and acceleration coefficients. But no one apply multiple modifications on more than one of the basic components of the PSO.

All survived papers which covered in this survey used only pure PSO. No one merge these modified techniques with one of computational intelligence techniques to get hybrid techniques and evaluate the performance of new techniques.

VIII. CONCLUSION

In this paper, we explain in details the main concepts of basic particle swarm optimization algorithm and its various modifications. Also, we present a review of stock market and we surveyed the most published works since 1998 and until 2014. Then we study if these different forms of PSO are implanted over the stock market.

As we see from table1, the most modification (about 90%) has been happened as the internal modification i.e. on the basic components of PSO, whereas about 45% of modifications has been happened on the velocity clamping, while only about 20% interested to make external modification on the PSO, and other of modifications has been applied over inertia term, lastly 25% of modifications has been applied on acceleration coefficients.

As future work, we can suggest the following points:

Up to our knowledge, all the modifications on the PSO were applied as only external or internal modifications, so a lot of work can be done if we apply external and internal modifications at the same time.

- In the internal modifications, all the modifications on the PSO were applied on one of the basic components of the PSO i.e. velocity clamping, inertia term, and acceleration coefficients, so a lot of work can be done if we apply multiple modifications on more than one of the basic components of the PSO.
- All techniques which covered in this survey used PSO, so a lot of work can be done if we combine these techniques with one of computational intelligence techniques such as genetic algorithm, bacterial foraging optimization, the Nearest Neighbor and the artificial neural networks. etc., to get hybrid techniques.
- Lastly, only few techniques which covered in this survey applied over stock market, so a lot of work can be done if we study the implementation of remind techniques over the stock market.

TABLE I. A SUMMARY OF MODIFICATIONS OF THE PSO

Author	year	Context	Description	External	Internal			Application
					Velocity Clamping	Inertia term	Acceleration Coefficients	Stock market
Shi et al	1998	LDWPSO	The inertia weight *w* is decreased linearly over the searching iterations, from an initial value to a final value			✓		✓
Van den Bergh and Engelbrecht	2002	GCPSO	Induce a new particle searching around the global best position found so far.		✓			
Xie et al.	2002	APSO	Improve swarm's local and global searching ability by inserting self-organization Theory.		✓			
He et al.	2004	PSOPC	Add a passive congregation part to the particle's velocity update formula.		✓			
Cui and Zeng	2004	SPSO	Particle i's position will be regenerated randomly if it is too close to the gbest.		✓			
Van den Bergh and Engelbrecht	2004	CPSO	Use multi-swarms to search different dimensions of the design space by employing cooperative behavior.		✓			
He and Han	2006	PSO-DT	Induce a disturbance term to the velocity update formula.		✓			
Yang et al.	2007	DAPSO	Use dynamic inertia weight to modify the velocity update formula.			✓		
Liu et al	2007	Center PSO	Introducing a center particle to the LDWPSO. The centre particle is proposed explicitly to visit the centre of the swarm at every iteration.		✓			✓
Zhao et al	2008	DMS-PSO	In this method the whole of the population are divided into small sized swarm.	✓				
Li and Xiao	2008	MBPSO	This method instead of using single global best position (gbest) and personal best position (Pbest), it uses the multi gbest and multi Pbest.	✓				
Ghali et al	2008	EPSO	Itupdates the form of inertia weight formula.			✓		✓

Deep et al	2009	MPSO	modifying the formula of velocity The two terms of original velocity update formula were replaced by two new terms based on the linear combination.		✓			✓
Zhao et al	2010	DMS-PSO	This method generate new harmonies according to the current personal best solution and the nearer personal best solution is replaced with a new harmony with better fitness.	✓				
Jie et al	2010	MPSO	This method applied a mixed local search behavior modes and information exchange among subswarms.	✓				
Azadani et al	2010	CPSO	Constraint handling is based on particle ranking and uniform distribution.				✓	
Jiang et al.	2010	MSSE-PSO	Population is sampled randomly from the feasible space and partitioned into several sub-swarms.	✓				
Engelbrecht	2010	HPSO	particles are allowed to follow different search behaviors selected from behavior pool	✓				
Chuang et al	2011	C-Catfish PSO	introduced chaotic maps into catfish particle swarm optimization			✓		
Ufuk et al.	2012	FET	The PSO algorithm is modified to assign a best local guide to each particle by using minimum angular distance information.		✓			
AlrijadjisDjoewahir at el.	2012	PSO-NDW	A modified PSO with Nonlinear Decreasing inertia Weight (PSO-NDW) employs the strategy that nonlinearly decreases the value of inertia weight from a large value to a small value.			✓		
Jie He et al.	2013		It take the gradient descent method (BP algorithm) as a particle swarm operator embedded in particle swarm algorithm, and at the same time they use to attenuation wall (Damping) approach to make fly off the search area of the particles of size remain unchanged and avoid the local optimal solution.	✓				
Yingsheng Su et al.	2013	FPSO	Each particle in FPSO is attracted by its previous best particle and other particles (not the global best particle) selected by a fuzzy mechanism.		✓			✓
Yubao Liu et al.	2014	MPSO	This algorithm modifies the strategy of generating new position of particles. For each generation solution, the flight velocity of particles is removed. Whereas the new position of each particle is generated by using difference strategy.		✓			

REFERENCES

[1] J. Kennedy, J. and Eberhart, C., "Particle Swarm Optimization". Proceedings of the 1995 IEEE International Conference on Neural Networks, Australia, 1995, pp. 1942-1948.

[2] Y.Jiang, C. Liu, C. Huang and X. Wu, 2010. Improved particle swarm algorithm for hydrological parameter optimization. Appl. Math. Comput., 217: 3207-3215.

[3] J. Jie, W. Wang, C. Liu and B. Hou, 2010. Multiswarm particle swarm optimization based on mixed search behavior. Proceedings of the 5th IEEE Conference on Industrial Electronics and Applications, Jun. 15-17, IEEE Xplore Press, Taichung, pp: 605-610. DOI: 10.1109/ICIEA.2010.5517044.

[4] J. Li, and X. Xiao, 2008. Multi-swarm and multi-best particle swarm optimization algorithm. Proceeding of the 7th World Congress on Intelligent Control and Automation, June 25-27, IEEE Xplore Press, Chongqing, pp: 6281-6286. DOI: 10.1109/WCICA.2008.4593876.

[5] R. Majhi, Panda, G. Majhi, B. and Sahoo, G., "Efficient prediction of stock market indices using adaptive bacterial foraging optimization (ABFO) and BFO based techniques", Expert Systems with Applications. Vol. 36(6), 2009, pp. 10097-10104.

[6] J. Li, and X. Xiao, 2008. Multi-swarm and multi-best particle swarm optimization algorithm. Proceeding of the 7th World Congress on Intelligent Control and Automation, June 25-27, IEEE Xplore Press, Chongqing, pp: 6281-6286. DOI: 10.1109/WCICA.2008.4593876

[7] Y. Shi, and Eberhart, R.C. "Parameter selection in particle swarm optimization", Proceedings of the 7th International Conference on Evolutionary Programming VII, March 25-27, 1998, p.591-600.

[8] R. Hassan, Nath, B. and Kirley, M., "A fusion model of HMM, ANN and GA for stock market forecasting", Expert Systems with Applications, Vol. 33, 2007, pp. 171-180.

[9] T.Aboueldahab, and Fakhreldin, M, "Stock Market Indices Prediction via Hybrid Sigmoid Diagonal Recurrent Neural Networks and Enhanced Particle Swarm Optimization", International Congress for global Science and Technology, ICGST, Vol. 10, 2010, pp. 23-30.

[10] G. Maddala, S., "Introduction to econometrics". New York, Toronto: Macmillan Publishing Company, 1992.

[11] T. László Kóczy, Claudiu R. Pozna, "Issues and Challenges of Intelligent Systems and Computational Intelligence", New York: Springer, | ISBN-10: 3319032054,2014 .

[12] K. Deep, and Bansal, J.C. "Mean particle swarm optimization for function optimization," Int. J. Computational Intelligence Studies, Vol. 1, No. 1, 2009, pp.72-92

[13] A. Engelbrecht, 2010. Heterogeneous particle swarm optimization. Proceeding of the 7th International Conference on Swarm Intelligence, pp: 191-202.

[14] N. I. Ghali, El-Desouki, N., Mervat, A. N. and Bakrawi, L., "Exponential Paticle Swarm Optimization Approach for Improving Data Clustering", Proc. World Academy of Science, Engineering and Technology, Vol.32, 2008, pp. 56-60.

[15] Yubao Liu, Guihe Qin, "A Modified Particle Swarm Optimization Algorithm for Reliability Redundancy Optimization Problem " in *Journal of Computers*, Vol 9, No 9 (2014), 2124-2131, Sep 2014, doi:10.4304/jcp.9.9.2124-2131.

[16] Y. Liu, Qin, Z., Shi, Z. and Lu, J., "Center particle swarm optimization," Neurocomputing Vol. 70, 2007, pp. 672-679.

[17] T. Helstrom, and Holmstrom, K. "Predicting the stock market". Published as Opuscula ISRN HEV-BIB-OP-26-SE. 1998.

[18] J. He , H. Guo, "A Modified Particle Swarm Optimization lgorithm ", in TELKOMNIKA Indonesian Journal of Electrical Engineering, Vol. 11, No. 10, October 2013, pp: 6209 - 6215 ,e-ISSN: 2087-278X

[19] R C. Eberhart, Simpson, P.K.and Dobbins, R.W., "Computational Intelligence PC Tools". Academic Press Professional, first edition, 1996.

[20] B. Malkiel, G., "A random walk down wall street". New York, London: W. W. Norton & Company, 1999.

[21] J. Kennedy, J. and Eberhart, C., "Particle Swarm Optimization". Proceedings of the 1995 IEEE International Conference on Neural Networks, Australia, 1995, pp. 1942-1948.

[22] S. He, Q. Wu, J. Wen, J. Saunders and R. Paton, 2004. A particle swarm optimizer with passive congregation.Biosystems, 78: 135-147.

[23] Z. Cui, and J. Zeng, 2004. A guaranteed global convergence particle swarm optimizer. Proceedings of 4th International Conference on Rough Sets and Current Trends in Computing, Uppsala, Sweden, pp: 762-767.

[24] U.¨ OZKAYA, F. G¨ UNES"A modified particle swarm optimization algorithm and its application to the multiobjective FET modeling problem", in Elec Eng & Comp Sci journal, Vol.20, No.2, 2012.

[25] Q. He, and C. Han, 2006. An improved particle swarm optimization algorithm with disturbance term. Comput. Intell. Bioinfo., 4115: 100-108.

[26] Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola , "Kernel methods in machine learning", in The Annals of Statistics Journal, Volume 36, Number 3 (2008), 1171-1220.

[27] X. Yang, J. Yuan and H. Mao, 2007. A modified particle swarm optimizer with dynamic adaptation. Appl. Math. Comput., 189: 1205-1213.

[28] S. Yingsheng and H. Fan,"AN IMPROVED FUZZY PARTICLE SWARM OPTIMIZATION FOR NUMERICAL OPTIMIZATION",in Dynamics of Continuous, Discrete and Impulsive Systems vol. 20, no. 2, pp. 173-188,2013.

[29] W.J. Xie, Zhang and Z.L. Yang, 2002a. Adaptive particle swarm optimization on individual level. 6th International Conference on Signal Processing, pp:1215-1218.

[30] L.Y. Chuang, S.W. Tsai and C.H. Yang, 2011. Chaotic catfish particle swarm optimization for solving global numerical optimization problems. Appl. Math. Comput., 217: 6900-6916.

[31] A. Djoewahir, T. Kanya, and M. Shenglin,"A Modified Particle Swarm Optimization with Nonlinear Decreasing Inertia Weight Based PID Controller for Ultrasonic Motor" International Journal of Innovation, Management and Technology, Vol. 3, No. 3, June 2012.

[32] M.Clerc, and J. Kennedy, "The particle swarm-explosion, stability and convergence in a multi dimensional complex space," IEEE Transaction Evolutionary Computation, Vol. 6, 2002, pp.58–73.

[33] E.N., S. Hosseinian and B. Moradzadeh, 2010. Generation and reserve dispatch in a competitive market using constrained particle swarm optimization. Int. J. Electr. Power Energ. Syst. 32:79-86.

A survey on top security threats in cloud computing

Muhammad Kazim
University of Derby
Derby, United Kingdom

Shao Ying Zhu
University of Derby Derby,
United Kingdom

***Abstract*—Cloud computing enables the sharing of resources such as storage, network, applications and software through internet. Cloud users can lease multiple resources according to their requirements, and pay only for the services they use. However, despite all cloud benefits there are many security concerns related to hardware, virtualization, network, data and service providers that act as a significant barrier in the adoption of cloud in the IT industry. In this paper, we survey the top security concerns related to cloud computing. For each of these security threats we describe, i) how it can be used to exploit cloud components and its effect on cloud entities such as providers and users, and ii) the security solutions that must be taken to prevent these threats. These solutions include the security techniques from existing literature as well as the best security practices that must be followed by cloud administrators.**

***Keywords*—Cloud computing; Data security; Network security; Cloud service provider security**

I. Introduction

Cloud computing offers many advantages such as increased utilization of hardware resources, scalability, reduced costs, and easy deployment. As a result, all the major companies including Microsoft, Google and Amazon are using cloud computing. Moreover, the number of customers moving their data to cloud services such as iCloud, Google Drive, Dropbox, Facebook and LinkedIn are increasing every day.

Many business level security policies, standards, and practices cannot be implemented in cloud due to which different security risks arise. Although cloud security has been a focused area of research in the last decade, there are still open challenges in achieving it. To control the security risks in cloud, it is crucial for researchers, developers, service providers, and users to understand them so that they can take maximum precautions, deploy existing security techniques or develop new ones. In this paper, the top security threats for cloud computing presented by Cloud Security Alliance (CSA) [1] have been analyzed.

The CSA guide [1] presents the security threats for cloud in the order of their severity and provides controls that can be followed by the service providers to avoid these threats. However, these threats and the controls to avoid them are very mentioned specifically to meet the requirements of industry. Therefore, there is a need to survey the security threats for cloud and their solutions from the research perspective. In this paper we define these threats, describe the ways they can be launched in cloud, the possible ways to exploit these threats and their effects on cloud entities. We have comprehensively analyzed and presented the security solutions for the prevention of these threats from literature. Moreover, we have classified these security issues into three categories which are data security, network security and cloud environment security (that includes issues specific to cloud environment).

This paper is composed as follows: Section II describes the most critical threats for cloud computing and their effects on cloud entities. Section III describes the security solutions to avoid these threats, and section IV gives the conclusion of paper.

II. Threats in Cloud Computing

In this section the major threats for cloud computing are explored. These are: i) data threats including data breaches and data loss, ii) network threats including account or service hijacking, and denial of service, and iii) cloud environment specific threats including insecure interfaces and APIs, malicious insiders, abuse of cloud services, insufficient due diligence, and shared technology vulnerabilities.

A. Data Threats

Data is considered to be one the most important valuable resource of any organization and the number of customers shifting their data to cloud is increasing every day. Data life cycle in cloud comprises of data creation, transit, execution, storage and destruction. Data may be created in client or server in cloud, transferred in cloud through network and stored in cloud storage. When required data is shifted to execution environment where it can be processed. Data can be deleted by its owner to complete its destruction.

The biggest challenge in achieving cloud computing security is to keep data secure. The major issues that arise with the transfer of data to cloud are that the customers don't have the visibility of their data and neither do they know its location. They need to depend on the service provider to ensure that the platform is secure, and it implements necessary security properties to keep their data safe.

The data security properties that must be maintained in cloud are confidentiality, integrity, authorization, availability and privacy. However, many data issues arise due to improper handling of data by the cloud provider. The major data security threats include data breaches, data loss, unauthorized access, and integrity violations. All of these issues occur frequently on cloud data. In this paper, we focus on data breaches and data loss that are described as the two most severe threats to cloud computing by CSA [1].

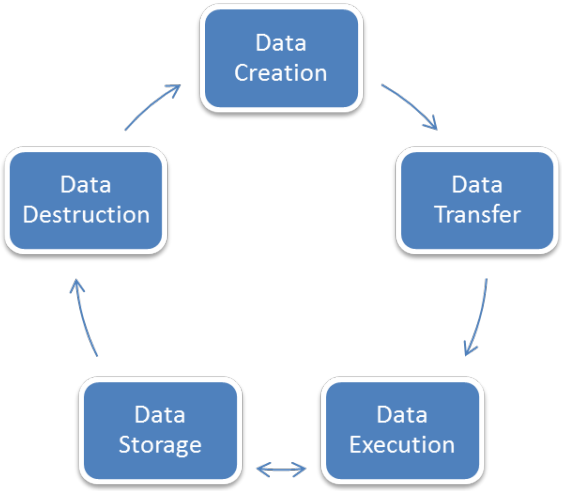


Fig. 1: Data life cycle in cloud computing

1) Data Breaches: Data breach is defined as the leakage of sensitive customer or organization data to unauthorized user. Data breach from organization can have a huge impact on its business regarding finance, trust and loss of customers. This may happen accidently due to flaws in infrastructure, application designing, operational issues, insufficiency of authentication, authorization, and audit controls [2]. Moreover, it can also occur due to other reasons such as the attacks by malicious users who have a virtual machine (VM) on the same physical system as the one they want to access in unauthorized way.

Apple's iCloud users faced a data leakage attack recently in which an attempt was made to gain access to their private data. Such attacks have also been done at other companies cloud such as Microsoft, Yahoo and Google. An example of data breach is cross VM side channel attack introduced by Y. Zhang et al., that extracts cryptographic keys of other VMs on the same system and can access their data [3].

2) Data Loss: Data loss is the second most important issue related to cloud security. Like data breach, data loss is a sensitive matter for any organization and can have a devastating effect on its business. Data loss mostly occurs due to malicious attackers, data deletion, data corruption, loss of data encryption key, faults in storage system, or natural disasters. 44 percent of cloud service providers have faced brute force attacks in 2013 that resulted in data loss and data leakage [4]. Similarly, malware attacks have also been targeted at cloud applications resulting in data destruction.

B. Network Threats

Network plays an important part in deciding how efficiently the cloud services operate and communicate with users. In developing most cloud solutions, network security is not considered as an important factor by some organizations. Not having enough network security creates attacks vectors for the malicious users and outsiders resulting in different network threats. Most critical network threats in cloud are account or service hijacking, and denial of service attacks.

1) Account or Service Hijacking: Account hijacking involves the stealing of user credentials to get an access to his account, data or other computing services. These stolen credentials can be used to access and compromise cloud services. The network attacks including phishing, fraud, Cross Site Scripting (XSS), botnets, and software vulnerabilities such as buffer overflow result in account or service hijacking. This can lead to the compromise of user privacy as the attacker can eavesdrop on all his operations, modify data, and redirect his network traffic. 1n 2009 a legitimate service was purchased from Amazon's EC2, and compromised to act as Zeus botnet [5].

2) Denial of Service: Denial of Service (DOS) attacks are done to prevent the legitimate users from accessing cloud network, storage, data, and other services. DOS attacks have been on rise in cloud computing in past 5 years and 81 percent customers consider it as a significant threat in cloud [1]. They are usually done by compromising a service that can be used to consume most cloud resources such as computation power, memory, and network bandwidth. This causes a delay in cloud operations, and sometimes cloud is unable to respond to other users and services.

Distributed Denial of Service (DDOS) attack is a form of DOS attacks in which multiple network sources are used by the attacker to send a large number of requests to the cloud for consuming its resources. It can be launched by exploiting the vulnerabilities in web server, databases, and applications resulting in unavailability of resources.

C. Cloud environment specific threats

Cloud service providers are largely responsible for controlling the cloud environment. However, a survey report by Alert Logic [4] shows that almost 50 percent of the cloud users consider service provider issues as a major threat in cloud computing. Apart from service provider threats, some threats are specific to cloud computing such as providing insecure interfaces and APIs to users, malicious cloud users, shared technology vulnerabilities, misuse of cloud services,

and insufficient due diligence by companies before moving to cloud.

1) Insecure Interfaces and APIs: Application Programming Interface (API) is a set of protocols and standards that define the communication between software applications through internet. Cloud APIs are used at all the infrastructure, platform and software service levels to communicate with other services. Infrastructure as a Service (IaaS) APIs are used to access and manage infrastructure resources including network and VMs, Platform as a Service (PaaS) APIs provide access to the cloud services such as storage and Software as a Service (SaaS) APIs connect software applications with the cloud infrastructure. The security of various cloud services depends on the APIs security. Weak set of APIs and interfaces can result in many security issues in cloud. Cloud providers generally offer their APIs to third party to give services to customers. However, weak APIs can lead to the third party having access to security keys and critical information in cloud. With the security keys, the encrypted customer data in cloud can be read resulting in loss of data integrity, confidentiality and availability. Moreover, authentication and access control principles can also be violated through insecure APIs.

2) Malicious Insiders: A malicious insider is someone who is an employee in the cloud organization, or a business partner with an access to cloud network, applications, services, or data, and misuses his access to do unprivileged activities. Cloud administrators are responsible for managing, governing, and maintaining the complete environment. They have access to most data and resources, and might end up using their access to leak that data. Other categories of malicious insiders involve hobbyist hackers who are administrators that want to get unauthorized sensitive information just for fun, and corporate espionage that involves stealing secret information of business for corporate purposes that might be sponsored by national governments.

3) Abuse of Cloud Services: The term abuse of cloud services refers to the misuse of cloud services by the consumers. It is mostly used to describe the actions of cloud users that are illegal, unethical, or violate their contract with the service provider. Abusing of cloud services was considered to be the most critical cloud threat in 2010 [2], and different measures were taken to prevent it. However, 84 percent of cloud users still consider it as a relevant threat [1]. Research has shown that some cloud providers are unable to detect attacks launched from their networks, due to which they are unable to generate alerts or block any attacks. The abuse of cloud services is a more serious threat to the service provider than service users. For instance, the use of cloud network addresses for spam by malicious users has resulted in blacklisting of all network addresses, thus the service provider must ensure all possible measures for preventing these threats.

Over the years, different attacks have been launched through cloud by the malicious users. For example, Amazon's EC2 services were used as a command and control servers to launch Zeus botnet in 2009 [6]. Famous cloud services such as Twitter, Google and Facebook as a command and control servers for launching Trojans and botnets. Other attacks that have been launched using cloud are brute force for password cracking of encryption, phishing, performing DOS attack against a web service at specific host, Cross Site Scripting and SQL injection attacks.

4) Insufficient Due Diligence: The term due diligence refers to individuals or customers having the complete information for assessments of risks associate with a business prior to using its services. Cloud computing offers exciting opportunities of unlimited computing resources, and fast access due which number of businesses shift to cloud without assessing the risks associated with it.

Due to the complex architecture of cloud, some of organization security policies cannot be applied using cloud. Moreover, the cloud customers have no idea about the internal security procedures, auditing, logging, data storage, data access which results in creating unknown risk profiles in cloud. In some cases, the developers and designers of applications maybe unaware of their effects from deployment on cloud that can result in operational and architectural issues.

5) Shared Technology Vulnerabilities: Cloud computing offers the provisioning of services by sharing of infrastructure, platform and software. However, different components such as CPUs, and GPUs may not offer cloud security requirements such as perfect isolation. Moreover, some applications may be designed without using trusted computing practices due to which threats of shared technology arise that can be exploited in multiple ways. In recent years, shared technology vulnerabilities have been used by attackers to launch attacks on cloud. One such attack is gaining access to the hypervisor to run malicious code, get unauthorized access to the cloud resources, VMs, and customers data.

Xen platform is an open source solution used to offer cloud services. Xen hypervisors code creates local privilege escalation (in which a user can have rights of another user) vulnerability that can be launch guest to host VM escape attack. Later, Xen updated the code base of its hypervisor to fix that vulnerability. Other companies such as Microsoft, Oracle and SUSE Linux that were based on Xen also released updates of their software to fix the local privilege escalation vulnerability. Similarly, a report released in 2009 [7] showed the usage of VMware to run code from guests to hosts showing the possible ways to launch attacks.

III. Security techniques for threats protection

In this section the security methods to avoid the exploitation of threats mentioned in section II have been discussed. We describe the implementation of these security techniques at different levels to secure cloud from threats.

A. Data Security

1) Protection from Data Breaches: Various security measures and techniques have been proposed to avoid the data breach in cloud. One of these is to encrypt data before storage on cloud, and in the network. This will need efficient key management algorithm, and the protection of key in cloud. Some measures that must be taken to avoid data breaches in cloud are to implement proper isolation among VMs to prevent information leakage, implement proper access controls to prevent unauthorized access, and to make a risk assessment of the cloud environment to know the storage of sensitive data and its transmission between various services and networks.

Considerable amount of research has been carried out for the protection of data in cloud storage. CloudProof [8] is a system that can be built on top of existing cloud storages like Amazon S3 and Azure blob to ensure data integrity and confidentiality using encryption. To secure data in cloud storage attributed based encryption can be used to encrypt data with a specific access control policy before storage. Therefore, only the users with access attributes and keys can access the data [9]. Another technique to protect data in cloud involves using scalable and fine grained data access control [10]. In this scheme, access policies are defined based on the data attributes. Moreover, to overcome the computational overhead caused by fine grained access control, most computation tasks can be handed over to untrusted commodity cloud with disclosing data. This is achieved by combining techniques of attribute-based encryption, proxy re-encryption, and lazy re-encryption.

2) Protection from Data Loss: To prevent data loss in cloud different security measures can be adopted. One of the most important measure is maintain backup of all data in cloud which can be accessed in case of data loss. However, data backup must also be protected to maintain the security properties of data such as integrity and confidentiality. Different data loss prevention (DLP) mechanisms have been proposed in research and academics for the prevention of data loss in network, processing, and storage. Many companies including Symantec, McAfee, and Cisco have also developed solutions to implement data loss prevention across storage systems, networks and end points.

R Chow et al. proposed the usage of Trusted Computing to provide data security. A trusted server can monitor the functions performed on data by cloud server and provide the complete audit report to data owner. In this way, the data owner can be sure that the data access policies have not been violated [11]. Tomoyoshi T. et al. proposed a system to protect moving data of a company inside a USB even if it is lost. They also describe the protection of document in its complete life cycle and avoiding data loss through emails [12].

B. Network Security

1) Protection from Account or Service Hijacking: Account or service hijacking can be avoided by adopting different security features on cloud network. These include employing intrusion detection systems (IDS) in cloud to monitor network traffic and nodes for detecting malicious activities. Intrusion detection and other network security systems must be designed by considering the cloud efficiency, compatibility and virtualization based context [13]. An IDS system for cloud was designed by combining system level virtualization and virtual machine monitor (responsible for managing VMs) techniques [14]. In this architecture, the IDSs are based on VMs and the sensor connectors on Snort which is a well-known IDS [15]. VM status and their workload are monitored by IDS and they can be started, stopped and recovered at any time by management system of IDS.

Identity and access management should also be implemented properly to avoid access to credentials. To avoid account hijacking threats, multi factor authentication for remote access using at least two credentials can be used. A technique that uses multi-level authentication at different levels through passwords was made to access the cloud services. First the user is authenticated by the cloud access password and in the next level the service access password of user is verified [16]. Moreover, user access to cloud services and applications should be approved by cloud management. The auditing of all the privileged activities of the user along with information security events generated from it should also be done to avoid these threats [17].

2) Protection from Denial of Service: To avoid DOS attacks it is important to identify and implement all the basic security requirements of cloud network, applications, databases, and other services. Applications should be tested after designing to verify that they have no loop holes that can be exploited by the attackers.

The DDOS attacks can be prevented by having extra network bandwidth, using IDS that verify network requests before reaching cloud server, and maintaining a backup of IP pools for urgent cases. Industrial solutions to prevent DDOS attacks have also been provided by different vendors. C. Jin et al. [18] proposed a technique named hop count filtering that can be used to filter spoofed IP packets, and helps in decreasing DOS attacks by 90 percent. Another technique for securing cloud from DDOS involves using intrusion detection system in VM [19]. In this scheme when an IDS detects an abnormal increase in inbound traffic, the targeted applications are transferred to VMs hosted on another data center.

C. Cloud Environment Security

1) Protection from Insecure Interfaces and APIs: To protect the cloud from insecure API threats it is important for the developers to design these APIs by following the principles of trusted computing. Cloud providers must also ensure that all the all the APIs implemented in cloud are designed securely, and check them before deployment for possible flaws. Strong authentication mechanisms and access controls must also be implemented to secure data and services from insecure interfaces and APIs. The Open Web Application Security Project (OWASP) [20] provides standards and guidelines to develop secure applications that can help in avoiding such application threats. Moreover, it is the responsibility of customers to analyze the interfaces and APIs of cloud provider before moving their data to cloud.

2) Protection from Malicious Insiders: The protection from these threats can be achieved by limiting the hardware and infrastructure access only to the authorized personnel. The service provider must implement strong access control, and segregation of duties in the management layer to restrict administrator access to only his authorized data and software. Auditing on the employees should also be implemented to check for their suspicious behaviour. Moreover, the employee behaviour requirements should be made part of legal contract, and action should be taken against anyone involved in malicious activities [17]. To prevent data from malicious insiders encryption can also be implemented in storage, and public networks.

3) Protection from Abuse of Cloud Services: The implementation of strict initial registration and validation processes can help in identifying malicious consumers. The policies for the protection of important assets of organization must also

be made part of the service level agreement (SLA) between user and service provider. This will familiarize user about the possible legal actions that can be conducted against him in case he violates the agreement. The Service Level Agreement definition language (SLAng) [21] enables to provide features for SLA monitoring, enforcement and validation. Moreover, the network monitoring should be comprehensive for detecting malicious packets and all the updated security devices in network should be installed.

4) Protection from Insufficient Due Diligence: It is important for organizations to fully understand the scope of risks associated with cloud before shifting their business and critical assets such as data to it. The service providers must disclose the applicable logs, infrastructure such as firewall to consumers to take measures for securing their applications and data [17]. Moreover, the provider must setup requirements for implementing cloud applications, and services using industry standards. Cloud provider should also perform risk assessment using qualitative and quantitative methods after certain intervals to check the storage, flow, and processing of data.

5) Protection from Shared Technology Vulnerabilities: In cloud architecture, hypervisor is responsible for mediating interactions of virtual machines and the physical hardware. Therefore, hypervisor must be secured to ensure proper functioning of other virtualization components, and implementing isolation between VMs. Moreover, to avoid shared technology threats in cloud a strategy must be developed and implemented for all the service models that includes infrastructure, platform, software, and user security. The baseline requirements for all cloud components must be created, and employed in design of cloud architecture. The service provider should also monitor the vulnerabilities in the cloud environment, and release patches to fix those vulnerabilities regularly [17].

IV. CONCLUSION

Cloud computing is getting widely adopted in businesses around the world. However, there are different security issues associated with it. In order to maintain the trust of customers, security should be considered as an integral part of cloud. In this paper we have focused on most severe threats on cloud computing that are considered relevant by most users and businesses. We have divided these threats into categories of data threats, networks threats, and cloud environment specific threats. The impact of these threats on cloud users and providers has been illustrated in the paper. Moreover, we also discuss the security techniques that can be adopted to avoid these threats.

REFERENCES

[1] T. T. W. Group *et al.*, "The notorious nine: cloud computing top threats in 2013," *Cloud Security Alliance*, 2013.

[2] C. S. Alliance, "Top threats to cloud computing v1. 0," *Cloud Security Alliance, USA*, 2010.

[3] Y. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Cross-vm side channels and their use to extract private keys," in *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 2012, pp. 305–316.

[4] "Cloud security report spring 2014," https://www.alertlogic.com/resources/cloud-security-report//, last Accessed: 2014-11-08.

[5] "Zeus bot found using amazon's ec2 as c and c server," http://www.theregister.co.uk/2009/12/09/amazon_ec2_bot_control_channel//, last Accessed: 2014-11-15.

[6] "Amazon ec2 cloud service hit by botnet, outage," http://www.cnet.com/uk/news/amazon-ec2-cloud-service-hit-by-botnet-outage/, last Accessed: 2014-11-15.

[7] K. Kortchinsky, "Cloudburst: A vmware guest to host escape story," *Black Hat USA*, 2009.

[8] R. A. Popa, J. R. Lorch, D. Molnar, H. J. Wang, and L. Zhuang, "Enabling security in cloud storage slas with cloudproof." in *USENIX Annual Technical Conference*, vol. 242, 2011.

[9] S. Ruj, M. Stojmenovic, and A. Nayak, "Decentralized access control with anonymous authentication of data stored in clouds," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 2, pp. 384–394, 2014.

[10] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *INFOCOM, 2010 Proceedings IEEE*. Ieee, 2010, pp. 1–9.

[11] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, "Controlling data in the cloud: outsourcing computation without outsourcing control," in *Proceedings of the 2009 ACM workshop on Cloud computing security*. ACM, 2009, pp. 85–90.

[12] T. Takebayashi, H. Tsuda, T. Hasebe, and R. Masuoka, "Data loss prevention technologies," *Fujitsu Scientific and Technical Journal*, vol. 46, no. 1, pp. 47–55, 2010.

[13] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.

[14] S. Roschke, F. Cheng, and C. Meinel, "Intrusion detection in the cloud," in *Dependable, Autonomic and Secure Computing, 2009. DASC'09. Eighth IEEE International Conference on*. IEEE, 2009, pp. 729–734.

[15] "Snort," https://www.snort.org//, last Accessed: 2015-01-29.

[16] H. Dinesha and V. Agrawal, "Multi-level authentication technique for accessing cloud services," in *Computing, Communication and Applications (ICCCA), 2012 International Conference on*. IEEE, 2012, pp. 1–4.

[17] "Cloud controls matrix (ccm), cloud security alliance," https://cloudsecurityalliance.org/research/ccm/, last Accessed: 2014-12-02.

[18] C. Jin, H. Wang, and K. G. Shin, "Hop-count filtering: an effective defense against spoofed ddos traffic," in *Proceedings of the 10th ACM conference on Computer and communications security*. ACM, 2003, pp. 30–41.

[19] A. Bakshi and B. Yogesh, "Securing cloud from ddos attacks using intrusion detection system in virtual machine," in *Communication Software and Networks, 2010. ICCSN'10. Second International Conference on*. IEEE, 2010, pp. 260–264.

[20] D. Fox, "Open web application security project," *Datenschutz und Datensicherheit-DuD*, vol. 30, no. 10, pp. 636–636, 2006.

[21] A. Al Falasi and M. A. Serhani, "A framework for sla-based cloud services verification and composition," in *Innovations in Information Technology (IIT), 2011 International Conference on*. IEEE, 2011, pp. 287–292.

Allocation of Roadside Units for Certificate Update in Vehicular Ad Hoc Network Environments

Sheng-Wei Wang
Department of Applied Informatics
Fo Guang University
Yilan 26247, TAIWAN

***Abstract*—The roadside unit (RSU) plays an important role in VANET environments for privacy preservation. In order to conserve the privacy of a vehicle, the issued certificate must be updated frequently via RSUs. If a certificate expires without being updated, the services for the vehicle will be terminated. Therefore, deploying as more as possible RSUs ensures that the certificate can be updated before it expires. However, the cost for allocating an RSU is very high. In this paper, we consider the roadside unit allocating problem such that the certificates can be updated before it expired. Previous researches focus on the roadside unit placement problem in a small city in which for any origination-destination pair the certificate is limited to update at most once. The RSU placement problem in which more than once certificate updates are required is discussed in this paper. The RSU allocation problem is formulated and the decision problem of the RSUs allocation problem is proved as an NP-complete problem. We proposed three roadside unit placement algorithms which works well for a large city. In order to reduce the number of required RSUs for certificate update, we also proposed three backward removing methods to remove the intersections found by the RSU allocation methods. Simulation results show that the proposed algorithms yields lower number of required RSUs than the simple method named the most driving routes first method. One backward removing method named the least driving routes first backward removing method was shown to be able to further reduce the number of required RSUs.**

***Keywords*—Roadside units allocation, VANET, certificate update, privacy conservation, NP-complete**

I. Introduction

A large number of services such as driving route planning, on-line maps and instant accident notifications require the vehicles to communicate with each other or to connect to the Internet [2]. The number of such services is still increasing. With the growing number of the services, vehicular ad hoc networks(VANETs) are used as the infrastructure of service platform [3].

VANET is an instantiation of mobile ad hoc networks (MANETs) [3]. The main difference between VANETs and MANETS is the components of the networks. In MANETs, the mobile nodes move randomly and no fixed base station is established. However, VANETs consist of vehicles and a number of fixed roadside units (RSUs) to support message exchange. A typical VANET includes three major components, namely, trust authority(TA), on-board units(OBUs) and the roadside units(RSUs) [2]. Fig. 1 shows the relationships between TA, OBUs and RSUs in VANET environments. The functions of the three components are described as follows:

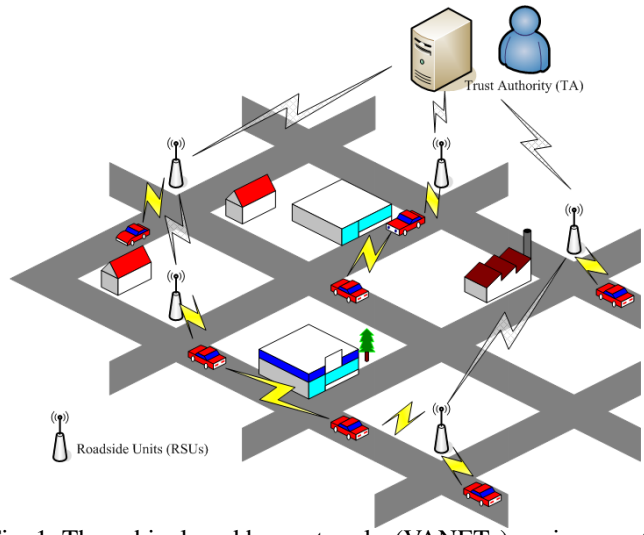


Fig. 1: The vehicular ad hoc networks (VANETs) environments

- Trust Authority (TA): The trust authority is a server which is managed by a service provider or the government. The function of a trust authority is to maintain the service, to keep the records of each vehicles or to issue the certificate for each vehicles.
- On-Board Unit (OBU): The on board unit(OBU) is equipped on a vehicle for inter-vehicles communications or communications between the vehicle and roadside units. An antenna is equipped in an OBU such that the vehicle communications with each other or the roadside units can be made.
- Roadside Unit (RSU): The roadside units are deployed on the traffic signs or along the roads. The main function of the roadside units is to bypass the messages between the vehicles and trust authority. Even though the function provided by the RSU is simple, RSU is a very important component in the VANET environments. If the number of RSU is small or if the RSUs are allocated inappropriately, the performances of the VANETs will become degraded.

In VANET environments, privacy conservation is an important issue in providing services to vehicles [4]. A number of mechanisms for conserving privacy have been proposed in [4]–[8]. Among the mechanisms, using certificate to identify the

owner of a message is an efficient way for secure service providing [7], [8]. However, if a certificate can not be updated for a long time, the certificate may be stolen by a potential eavesdropper. In order to prevent the certificate from being stolen, the certificate of each vehicle should be updated frequently. If the certificate can be updated more frequently, the services in which the authentication is made via the certificate will be more secure.

In VANET environments, a certificate update request is sent to trust authority via roadside units. The roadside unit receiving the certificate update request will pass the request to the trust authority. If the certificate is valid, the authority will issued a new certificate to the vehicle. The certificate which is no longer used will be put into the Certificate Revocation Lists (CRL) which disable the validity of the certificate [9]. When the vehicle receives a new certificate, the services can be provided to the vehicle continuously.

Although the RSU is important for certificate update, the cost for allocating an RSU is very high. To save the cost, only a small number of RSUs can be deployed in a city. How to allocate the RSUs in the VANET environment such that some objectives can be optimized is referred to as the **RSU allocation problem**. In this paper, we consider the problem of deploying a small number of RSUs such that the certificate can be efficiently updated in this city without expiration of a certificate.

In this paper, we directly show that the decision problem of RSU allocation problem is NP-complete. Three allocation methods, namely, *the most driving routes first* and *the most satisfied intersection pairs first*, and *the critical intersections first* methods, are proposed to find the locations for RSU deployment. Since the proposed algorithms are greedy based algorithms, we also proposed three backward removing methods to remove the some RSUs in the solutions found by the proposed RSU allocation methods. The proposed methods can be applied in a large city or under a short certificate updating interval environment. Simulation results show that our proposed the most satisfied intersection pairs first method and the critical intersections first method both yield lower number of required RSUs than the most driving routes first algorithm. We also show that one backward removing method named the least driving routes first method performs better than the other two backward removing methods.

The rest of this paper is as follows. Section II studies the related works in RSU allocation problems. In Section III, the RSU allocation problem is formulated and the NP-completeness of the problem is proved. Section IV provides three allocating methods and three backward removing methods for RSU allocation. Simulation results are shown and discussed in Section V. Finally, some concluding remarks are given.

II. Related Works

A number of researches have been proposed to find the placement for RSUs in the VANET environments [4], [10], [11]. In [11], the authors proposed an analytical model to estimate the minimum number of required RSUs where the packet delay between the vehicles and RSUs are bounded. The RSU placement problem which minimizes the disconnection time and maximizes the connectivity between the vehicles and RSUs are studied in [10] and a placement scheme was proposed.The RSU allocation problems focused in [11] and [10] are not the same as the problem focused in this paper.

In [4], the RSU problem for certificate update is studied and is transformed into a set cover problem. In [4], the transformed problem is proved as an NP-Hard problem and a greedy algorithm is proposed. However, since the instances transformed from RSU placement problem to the set cover problem are not proved as the general cases for set cover problem, the fact that the optimal set cover problem is NP-hard does not ensure that the RSU placement algorithm is NP-hard [12]. Besides, the algorithm proposed in [4] only applies to a small city in which only at most once certificate updating is allowed when driving along every shortest path between an origination-destination pair. Therefore, the authors do not take a large city or a short certificate valid interval into account.

TABLE I: Table of Notations of RSU Allocation Problem

Notation	Definition
M	a graph represents a city map
I	the set of intersections in M
R	the set of roads in M
T	the set of driving times in M
r_{ij}	the road from intersection i to j
t_{ij}	the driving time from intersection i to j
(s,d)	a origination-destination pair with origination intersection s and destination intersection d
$P(s,d)$	the driving route from intersection s to d
$T(s,d)$	the total driving time from intersection s to d
$S(s,d,k)$	the kth segment between (s,d)
$N(s,d)$	the number of segments in driving route $P(s,d)$
$T(s,d,k)$	the driving time on segment $S(s,d,k)$
n_{sd}	the number of all segments between (s,d)
A_i	an indicator to indicate whether an RSU is allocated on intersection i
C_T	the length of certificate valid interval
N	the required number of RSUs in city M

III. The RSU Allocation Problem

In this section, we first formulated the RSU allocation problem. The notations used in this section are summarized in Table I. We also proved that the decision problem of RSU allocation problem is NP-complete in this section.

A. Problem Formulation

In the RSU allocation problem, a map of a city is denoted as $M = (I, R, T)$ where I is the set of the intersections, R is the set of roads, and T is the driving time on each road in this city. Each road in this city is directional. If there is a road from one intersection to another, there is also a road in the reverse direction. The driving time on both directions are not necessarily the same.

The road and the driving time from from intersection i to its neighborhood intersection j are denoted as the r_{ij} and t_{ij} respectively. Let (s,d) be denoted as the origination(source)-destination intersection pair from intersection s to intersection d. The driving route between intersections s and d is denoted as $P(s,d)$ and the driving time on $P(s,d)$ is denoted as $T(s,d)$.

We assume that the RSUs can only be allocated in the intersections since the RSUs are always deployed on the traffic

signs and the certificate can be updated only when the vehicle passed the intersections. When some RSUs are allocated in the city, each driving route in the city may be divided into several sub-routes, namely, segments. The endpoints of a segment is the origination intersection, destination intersection or the intersections with RSU allocated. Any non-endpoint in a segment is an intersection without RSU allocated. We let the segment $S(s,d,k)$ be the kth segment on driving route $P(s,d)$. The number of segments in driving route $P(s,d)$ is $N(s,d)$. The driving time on segment $S(s,d,k)$ is $T(s,d,k)$.

The indicator A_i indicates whether the intersection i is allocated an RSU or not. That is,

$$A_i = \begin{cases} 1 & \text{if an RSU is allocated at intersection } i, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The number of RSU allocated is denoted as N. That is,

$$N = \sum_{i \in I} A_i. \quad (2)$$

The length of a certificate valid interval is denoted as C_T which is a fixed value. To update the certificate before it expires, the length of the certificate valid interval should be longer than or equal to the driving time over each segment. That is,

$$C_T \geq T(s,d,k),\ \forall i,j \in I,\ i \neq j, \text{ and } k = 1,2,\cdots,N(s,d). \quad (3)$$

If driving time on a road r_{ij} is larger than C_T, the certificate can not be updated no matter how the RSUs are allocated. To ensure that the certificate can be updated on all driving route, we assume that the driving time on each road r_{ij}, t_{ij}, is less than or equal to the length of certificate valid interval C_T. That is,

$$C_T \geq t_{ij}, \quad \forall i,j \in I, \quad i \neq j. \quad (4)$$

The objective function for the RSU allocation problem is to minimize the number of required RSUs in the city such that the certificate can be updated before it expires on all driving routes.

The RSU allocation problem can be described as follows. *Given a city map $M = (I,R,T)$, the driving route $P(s,d)$ for all intersections s and d, and the length of certificate interval C_T, the objective is to find a subset of intersections A such that each segment $S(s,d,k)$ on $P(s,d)$, the driving time $T(s,d,k)$ is shorter than or equal to the length of certificate valid interval C_T and the number of intersections in A is minimized.*

B. NP-Completeness of The RSU Allocation Problem

To prove the NP-completeness of the RSU allocation problem, we transform the problem into a decision problem.

> *Given a city map $M = (I,R,T)$, the driving route $P(s,d)$ for all intersections s and d, and the length of certificate interval C_T, we want to find a subset of intersections A such that each segment $S(s,d,k)$ on $P(s,d)$, the driving time $T(s,d,k)$ is shorter than or equal to the length of certificate valid interval C_T and the number of intersections in A is less than or equal to N?*

TABLE II: Table of Notations for Hitting Set Problem

Notation	Definition
F	a finite set
f_i	a subset of F
e_i^j	the jth element in f_i after sorting
C	the set of f_i
F'	a hitting set for C
M'	corresponding map for hitting set problem
I'	corresponding set of intersections
R'	corresponding set of roads
T'	corresponding set of driving times
P_i'	the driving route in M' corresponding to f_i
P'	the set of driving routes in M'

We proved the decision problem of RSU allocation problem is NP-complete in the following theorem.

Theorem 1 (NP-Completeness): The RSU-ALLOCATION-PROBLEM is NP-complete.

Proof: First, we will show the RSU allocation problem is in NP. Given a set intersections on the city map, it is easy to check whether the allocations for RSU placement is sufficient for certificate update and the number of intersections is less than or equal to N. Time complexity of the verification is polynomial time where the computation time is proportional with the number of origination-destination pairs in the given city. A solution for the RSU allocation problem can be verified in polynomial time; that is, the RSU allocation problem is in NP.

In order to prove that the RSU allocation problem is NP-complete, we first find an existing NP-complete problem to reduced to the RSU allocation problem. We transform the HITTING SET problem [12] to the RSU-ALLOCATION-PROBLEM. The MINIMUM-HITTING-SET is described as follows [12].

> *HITTING-SET PROBLEM: Given a collection C of subsets of a finite set F, we want to find a hitting set for F, i.e., a subset $F' \subseteq F$ such that F' contains at least one element from each element in C and the cardinality of the hitting set, i.e., $|F'|$ is less than or equal to K.*

The notations used for HITTING SET problem is summarized in Table II.

We then construct the corresponding city maps $M' = (I',R',T')$ for an instance of the hitting set problem as follows. Let each intersection in I' represent each element in F and for each element f_i in C, add two intersections s_i and d_i into the set of intersections I'.

For each element f_i in C, sort the elements in f_i in ascending order first. The elements in f_i can be denoted as $f_i = \{e_i^1, e_i^2, \cdots, e_i^{k_i}\}$ where k_i is the number of elements in f_i. Note that if $a < b$, e_i^a is smaller than e_i^b. For each element e_i^j, $j = 1,2,\cdots,k_i - 1$ in sorted set f_i, add a road from intersection e_i^j to e_i^{j+1} and the driving time on the road is set to 0. Next, for each sorted set f_i, add two roads from s_i to e_i^1 and from $e_i^{k_i}$ to d_i with driving time T where T is an arbitrary positive real number. The driving routes P_i' can be constructed from each s_i to d_i along the intersections specified

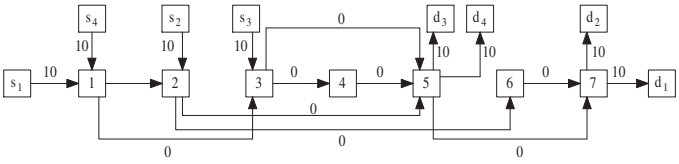


Fig. 2: Example for proof of Theorem 1

in f_i. Finally, we set the length of certificate valid interval C_T to $\frac{3T}{2}$ and set the value of N to the value of K.

For example, given a finite set $S = \{1, 2, 3, 4, 5, 6, 7\}$ and a collection $C = \{\{1, 3, 5, 7\}, \{2, 6, 7\}, \{3, 4, 5\}, \{1, 2, 5\}\}$, we construct the map as follows. The intersections from 1 to 7 are first constructed and the four pairs of intersections $s_1, d_1, s_2, d_2, s_3, d_3, s_4$ and d_4, are then generated. The set of driving routes P' corresponding to C consists of four components where $P'_1 = \{s_1, 1, 3, 5, 7, d_1\}$, $P'_2 = \{s_2, 2, 6, 7, d_2\}$, $P'_3 = \{s_3, 3, 4, 5, d_3\}$, and $P'_4 = \{s_4, 1, 2, 5, d_4\}$. The roads are added according to the set in collection C and the driving time are added on the roads. We let $T = 10$ and the $C_T = 15$. Fig. 2 shows the map generated after transformation.

For each driving route P'_i, it is obvious that the total driving time is $2T$. From intersection s_i to d_i, the driving time is larger than the length of certificate valid interval $C_T = \frac{3T}{2}$ such that the certificate is required to update at least once. Since the driving time is 0 on the roads not connected to intersection s_i or d_i on driving route P'_i, updating the certificate at each intersection on P'_i excluding s_i and d_i may made the certificate not expired during the driving time on P'_i. Finding an intersection on P'_i is equivalent to find an element in set f_i. Therefore, if the number of intersections in map M' such that certificate can be update before it expires on all driving routes is less than or equal to N, the cardinality of the hitting set for collection C is less than or equal to K. This completes the proof of Theorem 1. ■

IV. The RSU Allocation Methods

In this section, we first proposed three greedy based RSU allocation methods for certificate update. Since the proposed methods are all greedy based methods, the found allocations are not necessarily the optimal solution. We then proposed three backward removing methods to remove some locations of RSUs to obtain a better solution.

A. RSU allocation methods

We proposed three RSU allocation methods, namely, *the most driving routes first* and *the most satisfied intersection pairs first*, and *the critical intersections first* methods, in VANET environments. The three methods are described in the following.

1) The most driving routes first method: The idea of this method is that an intersection with more driving routes passed is more likely to become the location of RSU for certificate update. Hence, allocating an RSU on the intersection is expected to be effective for certificate update on the driving routes. Therefore, this method sorts the intersections in the city in descending order according to the number of driving routes passed the intersection and allocate the RSUs on the intersections one by one until the certificate can be update in time on all driving routes.

The computation time of this method includes the time for checking the number of traversed driving routes through each of the interconnection, the time for sorting the interconnections according the number of passed driving routes of the interconnections, and checking if the placed RSUs are enough for certificate update in this city. The computational complexity of counting the number of traversed driving routes is $O(|I|^2)$ since there are $|I| \times (|I| - 1)$ driving routes in the city, the computational complexity of sorting is $O(|I| \log |I|)$ and the computational complexity of checking if the RSUs allocated are enough for certificate update is $O(|I|)$. Therefore, the computational complexity of this method is $O(|I|^2)$.

2) The most satisfied intersection pairs first method: The idea of this method is as follow. If the location of an RSU on an intersection yields more source-destination pairs between which the certificate can be updated before it expires, it is more beneficial to allocate the RSU on the intersection. The details are described in the following.

Recall that $P(s, d)$ be the driving routes between source-destination intersection pair s and d. Let $A_\emptyset$ be the allocation pattern in which no RSU is allocated in the city. Let $h(A_\emptyset, s, d)$ be an indicator that whether the source-destination intersection pairs between which the certificate can be updated or not before it expires when no RSU is allocated in the city. Note that when no RSU is allocated in the city, only the source-destination intersection pair (s, d) with driving time $T(s, d)$ less than or equal to C_T satisfies the constraint described in equation (3). That is,

$$h(A_\emptyset, s, d) = \begin{cases} 1 & \text{if } T(s, d) \leq C_T, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The computational complexity for checking whether the driving time $T(s, d)$ is less than or equal to C_T or not is constant time, i.e. $O(1)$.

Let h be the total number of the source-destination pairs between which the certificate can be updated before it expires when no RSU is allocated in the city. Then h is calculated as follows:

$$h = \sum_{s,d \in I,\ s \neq d} h(A_\emptyset, s, d) \ . \tag{6}$$

The computational complexity for calculating h is $O(|I|^2)$.

Let $h(A_i, s, d)$ denote if the source-destination intersection pairs between which the certificate can be updated before it expires when no RSU is allocated in the city. Let $f(i)$ denote the total number of source-destination intersection pairs between which the certificate can be updated before it expires when a single RSU is allocated at intersection i. Then $f(i)$ is obtained as follows:

$$f(i) = \sum_{s,d \in I,\ s \neq d} h(A_i, s, d) \ . \tag{7}$$

The computational complexity for calculating each $f(i)$, is also $O(|I|^2)$. The computational complexity for calculating all $f(i),\ i \in I$, is $|I| \times O(|I|^2) = O(|I|^3)$.

Let $R(i)$ be the difference of the numbers of source-destination pairs which can be successfully updated when no RSU is allocated and when a single RSU is allocated at intersection i. Then $R(i)$ is given as follows:

$$R(i) = h - f(i) \ . \tag{8}$$

The intersection with highest $R(i)$ is selected for RSU allocation. When the first intersection is found and an RSU is allocated, the next intersection is found by the same procedure which can maximize the difference of the numbers of source-destination pairs between which the certificate can be successfully updated. The intersection with highest difference is selected. The procedure repeats until the certificate on all driving routes in the city can be updated in time. Since the procedure repeats at most $|I|$ times where each intersection us allocated an RSU, the overall computational complexity of this method is $|I| \times O(|I|^3) = O(|I|^4)$.

3) The critical intersections first method: Since the most satisfied intersection pairs first method requires high computational complexity, we devise the following method to reduce the computational time. We first find the *critical intersections* at which an RSU is required for certificate update in all feasible allocation patterns. The critical intersections can be found as follows. For each driving route with more than 2 roads, the non-endpoints on the driving route is checked if it is a critical intersection or not. For each consecutive roads r_{ij} and r_{jk}, intersection j is a critical intersection if the sum of driving time on the two roads, $t_{ij} + t_{jk}$, is larger than C_T. That is, if no RSU is allocated at intersection j, the certificate cannot be updated on the driving route. The computational finding the critical intersection is $O(|I|^3)$ since each intersection on all $|I| \times (|I| - 1)$ driving routes should be checked.

After finding the critical intersections, the RSUs are first allocated on the critical intersections. The most satisfied intersection pairs first method are then employed to find other intersections until the certificate on all driving routes can be updated successfully. The computational complexity of the most satisfied intersection pairs first method is $O(|I|^4)$ which is also the overall computational complexity.

Although the critical intersections first method has the same computational complexity as the most satisfied intersection pairs method, the computation time of the method is significantly less than that of the most satisfied intersection pairs first method under some values of C_T. The computation time of the critical intersections depends on how many critical intersections are found in the city. When C_T is not large, more critical intersections can be found in the first part of the method which reduced the search time. However, when C_T is large, only a few critical intersections can be found and the time for searching the other intersections is long.

B. Backward removing methods

Since there will be some redundant RSUs after allocating the RSUs by the three greedy based methods, we proposed three backward removing algorithms to remove the redundant RSUs. Note that the backward removing methods are applied when a set of intersections which is available for successfully certificate update in a city map is obtained by the RSU allocation methods. The three backward removing methods are described in the following.

1) The random backward removing method: Given a set of intersections which is available for successfully certificate update, the random backward removing method removes the intersections one by one in a random order. If an intersection can be removed such that the set of intersections after removing is also available for certificate update, the intersection will be removed. The random backward removing method removes the intersections one by one until all intersections are checked.

2) The most driving routes first backward removing method: The most driving routes first backward removing method works similar as the random method. The main difference between the two methods is in the list of intersections to be removed. The list for removing in the most driving routes first backward removing method is the same as the list in the most driving routes first RSU allocation method.

3) The least driving routes first backward removing method: The least driving routes first backward removing method works similar as the previous two methods. The list for removing in the least driving routes first backward removing method is the reverse list used in the most driving routes first backward removing method.

V. Simulation Study

Simulations are performed to study the performances of the proposed RSU allocation methods. First of all, the percentage of city maps in which the method proposed in [4] can apply successfully are discussed. The performances of the proposed methods are compared with each other. In addition, the locations found by the most driving routes first method are also investigated. We also study the the average driving time in each segment with different length of the certificate valid intervals. Finally, the performance of the three proposed backward removing methods are compared with the allocation method without backward removing method.

A. Simulation Model

Random city maps are used to represent the networks. The type of city maps is a square with 10 intersections each side. That is, the number of intersections in a city is 100. For each intersection i to a neighborhood intersection j, the driving time e_{ij} is selected from 20 to 100 with uniform distribution. The driving time from intersection i to j is not necessarily the same as that from intersection j to i. The number of origination-destination intersection pairs is 100×99.

For each simulation run, 100 random city maps are generated. Since most navigation systems use shortest path algorithm as the route planning algorithm [13], [14], the driving route between each source-destination pair is obtained by Dijkstra's shortest path algorithm [15] in the simulations. The length of certificate valid interval C_T ranges from 100 to 800.

For each city map, the numbers of required RSUs obtained by the allocation methods are first calculated. We also calculate the average length of segments to compare with the length of certificate valid interval C_T. Each data point in our graph is the average values over the 100 city maps.

In the figures to be presented in the following, the simulation results corresponding the most driving routes first method are labeled as *Driving Routes*. The most satisfied intersection

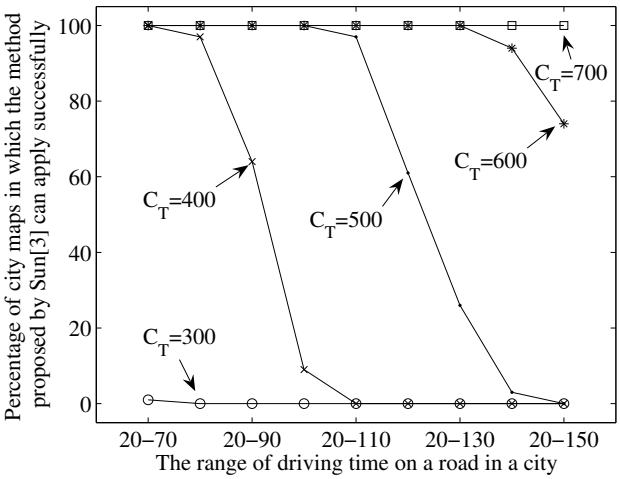


Fig. 3: Percentage of of city maps in which the method proposed in [4] for allocating RSUs such that the certificate can be updated before it expires

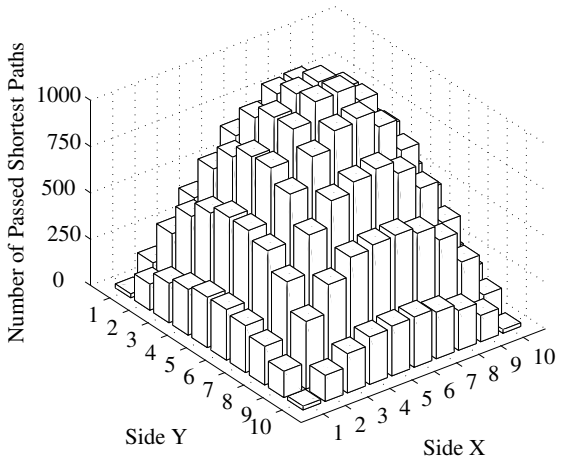


Fig. 4: The number of driving routes passed

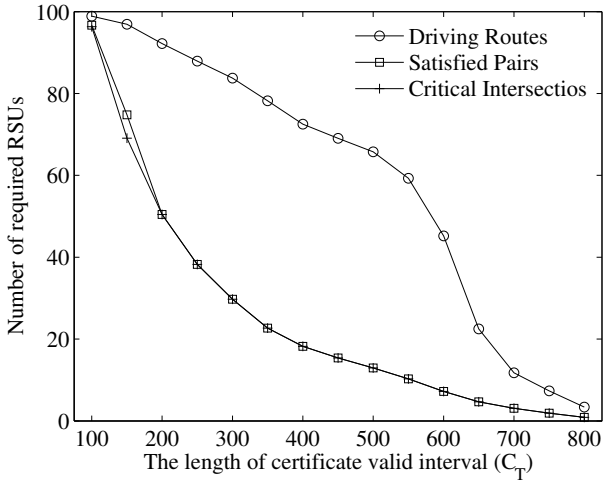


Fig. 5: The number of required RSUs in the city

pairs first method and the critical intersections first methods are labeled as *Satisfied Pairs* and *Critical Intersections* respectively. In the figures 7 and 8 , the allocation methods without backward removing methods are labeled as *No BR*. The three backward removing methods, the random backward removing method, the most driving routes first backward removing method, and the least driving routes first backward removing method are labeled as *Random*, *Most Driving Routes*, and *Least Driving Routes* respectively.

B. Simulation Results

First of all, we will show that the proposed methods are able to solve the RSU allocation problem in a large city or small certificate update interval compare with the proposed method in [4]. Figure 3 shows the percentage of the city maps in which the methods can successfully allocate the RSUs such that the certificate can be updated between all origination-destination pair among 100 city maps. From the figure, we can observe that the proposed methods achieves 100% success rate while success rate of the method in [4] decreased with the increasing certificate update interval.

Next, we are interested in some properties of the most driving routes first method performs when the driving routes are shortest paths between the source-destination pairs. The intersections are located in a square with 10×10 intersections. Fig. 4 shows the average number of shortest paths passed at each intersection from 100 city maps. From the figure, we found the following properties of the most driving routes first method:

- The intersections with minimum number of passed driving routes are located at the corners of the city. This is because if an intersection is at the corners or nearby, the number of shortest paths in the city passed the intersection is small.
- The intersections with maximum number of passed driving routes are located at the center of the city. When an intersection is located at the center of the city, many driving routes may traversed the intersection since the probability that the intersection is on the shortest path between a source-destination pair is high.
- The numbers of passed shortest paths of the nearby intersections are close. When an intersection is around the center of the city, the number of passed driving routes will be large. The value will be small when the intersection is near the boundary of the city.

The properties observed in fig. 4 are able to provide some explanations for the following simulation results.

We next compare the number of required RSUs obtained by the proposed methods. Fig. 5 shows the number of required RSUs with respect to different length of certificate valid intervals. From the figures, we can make the following observations:

- The numbers of required RSUs obtained by the three

methods decreased when the length of certificate valid interval increased. This result is trivial since that when the length of certificate valid interval is large, the need for certificate update will be less as well as the number of required RSUs in the city is small too.

- The number of RSUs obtained by the most driving routes is much larger than the other two methods. This is because that locations of the intersections with large number of passed driving routes are close. When the intersections are close, allocating the RSUs at the these locations will not be effective for certificate update. It is expected that allocating the RSUs more evenly in the city will yield lower number of required RSUs.
- The number of RSUs obtained by the most satisfied intersection pairs first method and the critical intersections first method are the same except when the length of certificate valid interval is smaller than 200. This is because that the maximum driving time on a road is set to 100 in this simulation such that it is impossible to find any critical intersection when C_T is set to larger than or equal to 200. When C_T is larger than or equal to 200, the critical intersections first method is equivalent to the most satisfied intersection pairs method because the critical intersections first method will not find any critical intersection in the first part of the method.
- When the certificate valid interval is smaller than 200, the critical intersections first method yields lower number of required RSUs than the most satisfied intersection pairs first method. Since the most satisfied intersection pairs first method does not find the critical intersections in the beginning, the result implies that some intersections which are able to maximize the number of satisfied intersection pairs but not a critical intersection will be found before some critical intersections. However, when the intersections are found, some of the non-critical intersections may be redundant.
- When the number of required RSUs increased, it is obvious that the average number of segments will decrease.

We are also interested in the average driving time on each segment compared with the length of certificate valid interval. Fig. 6 shows the results of the proposed methods. From the figure, we can made the following observations:

- The average driving time on each segment from the most driving routes first method is less than the other two methods. Since the number of allocated RSUs is large when the most driving routes first method is employed, the number of segment is also large which made the average driving time on each segment be small.
- The average driving time on each segment is much less than the length of certificate valid interval. Given a length of certificate valid interval C_T, it is expected that the driving time on all segments on all driving routes is less than C_T. Since all the driving time on a segment must be less than C_T, some RSUs are allocated for certificate update in a small number of driving routes. However, when the RSUs are allocated, the number of segments on all driving routes which go through the intersection will increase which further decreased the average driving time on each segment.

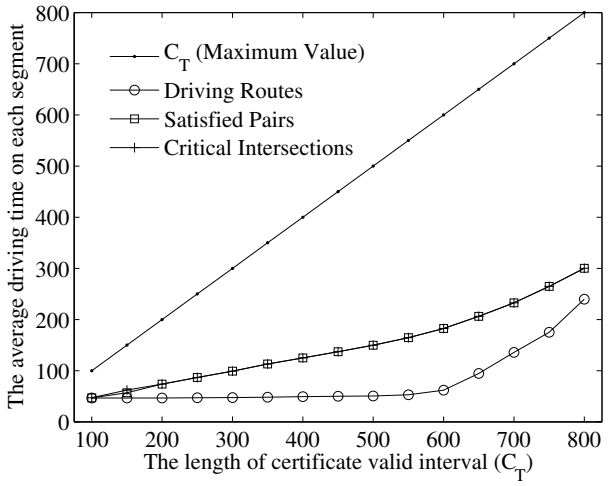


Fig. 6: The average driving time on each segment of driving route

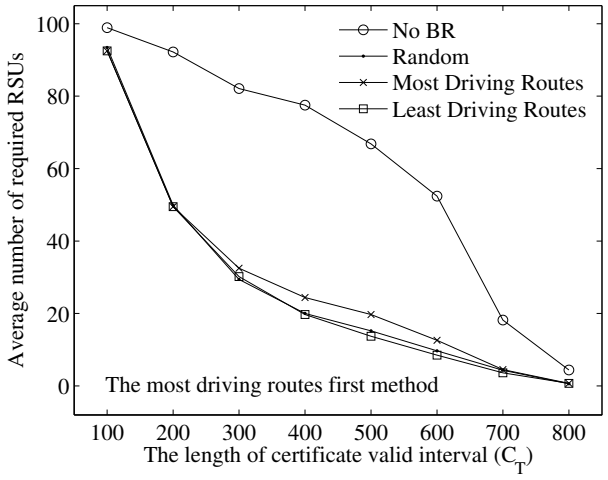


Fig. 7: The average number of required RSUs when the most driving routes first RSU allocation method is employed

Finally, we will discuss the performances of backward removing methods when the placement methods are the most driving routes first method and the most satisfied intersection pairs first method. The performances with three backward removing methods are compared with that without backward removing. Figures 7 and 8 show the average required RSUs when the most driving routes first method and the most satisfied intersection pairs first method are employed with the three backward removing methods.

From the figures, we can make the following observations:

- In Figure 7, the backward removing methods signifi-

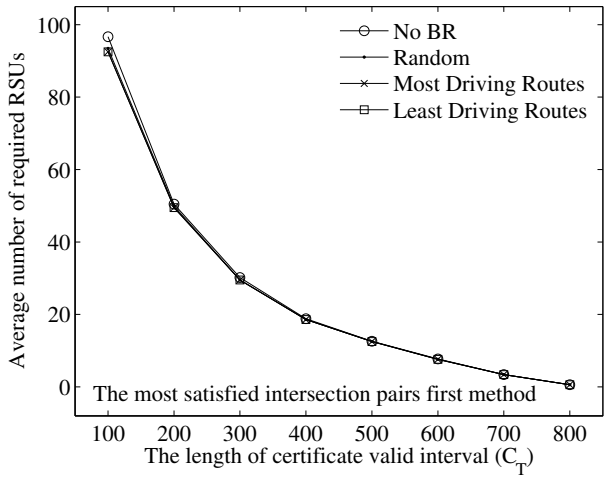


Fig. 8: The average number of required RSUs when the most satisfied intersection pairs first RSU allocation method is employed

cantly reduce the required RSUs compared with that without backward removing methods. This is because the most driving routes first RSUs allocation allocation method find too many redundant intersections for certificate update.

- In Figure 8, the performances of the proposed backward removing methods is similar as that without backward removing. The reason is that the most satisfied intersection pairs method finds the intersections which is indeed required and cannot be removed.
- Among the three backward removing methods, the least driving routes first method performs slightly better than other two methods.

VI. Conclusions

In this paper, we consider the roadside units allocation problem such that the certificates can be updated before it expired. The decision problem of the RSU allocation problem is shown as an NP-complete problem. We also proposed three RSU allocation algorithms which works for a large city. Simulation results show that one proposed method named the critical intersections first method yields lower number of required RSUs than the other two RSUs allocation method. We also show that a backward removing method named the least driving routes first method performs better than the other two backward methods if the RSU allocation method does not find good locations for roadside units. If the route planning algorithm rather than the shortest paths can be developed, the required number of RSUs can be further reduced. Developing the routing planning algorithms is left for the future researches.

Acknowledgements

This research was supported by the National Science Council, Taiwan, under grant NSC99-2218-E-431-001-MY3 .

References

[1] S.W. Wang & M.Y. Chang(2010). Roadside Units Allocation Algorithms for Certificate Update in VANET Environments. In *the 17th Asia-Pacific Conference on Communications (APCC 2011)*, Kota Kinabalu, Malaysia, October 2-5, 2011

[2] J. Zhao & G. Cao(2006). VADD: vehicle-assisted data delivery in vehicular ad hoc networks. In *IEEE INFOCOM 2006,* Barcelona, Spain.

[3] H. Hartenstein & K.P. Laberteaux(2010). *VANET: vehicular applications and inter-networking technologies,* John Wiley & Sons Inc.

[4] Y. Sun, X. Lin, R. Lu, X. Shen, & J. Su(2010). Roadside units deployment for efficient short-time certificate updating in VANETs. In *IEEE International Conferences on Communications (ICC) 2010,* Cape Town, South Africa.

[5] J.L. Huang, L.Y. Yeh, and H.Y. Chien(2011). ABAKA: an anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks. In *IEEE Transactions on Vehicular Technology,* 60(1), 248-262.

[6] S. Mahajan & A. Jindal(2010). Security and privacy in VANET to reduce authentication overhead for rapid roaming networks. In *International Journal of Computer Applications,* 1(20), 17-21.

[7] B. Aslam & C.C. Zou (2009). Distributed certificate architecture for VANETs. In *Proceedings of ACM SIGCOMM 2009,* , Barcelona, Spain.

[8] F. Dtzer(2006) Privacy issues in vehicular ad hoc networks . In *Privacy Enhancing Technologies,* Lecture Notes in Computer Science, 2006.

[9] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, & R. Housley, W. Polk (2008), Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile. In *IETF RFC 5280,* May 2008.

[10] J. Lee & C. Kim(2010). A roadside unit placement scheme for vehicular telematics networks. In *AST/UCMA/ISA/ACN 2010*, Miyazaki, Japan.

[11] A. Abdrabou & W. Zhuang (2011). Probabilistic delay control and road side unit placement for vehicular ad hoc networks with disrupted connectivity. In *IEEE Journal on Selected Areas in Communications,* 29(1), 129-139.

[12] M.R. Garey & D.S. Johnson(1979). *Computers and intractability: a guide to the theory of NP-completeness,* Bell Laboratories, Murray Hill, New Jersey.

[13] I. Flinsenberg(2009). In *Route planning algorithms for car navigation,* VDM Verlag, September 2009.

[14] C.C. Martin, P.R. Thrift, & M.C. Lineberry(1994). Systems and methods for planning the scheduling travel routes. U.S. Patent 5 272 638, December 21, 1994.

[15] E.W. Dijkstra(1959). A note on two problems in connexion with graphs. in *Numerische Mathematik 1*, 269-271.