Jan Zizka
Dhinaharan Nagamalai (Eds)

Computer Science & Information Technology

The Sixth International Conference on Computer Science, Engineering and Information Technology (CCSEIT 2016)
Vienna, Austria, May 21~22, 2016

AIRCC Publishing Corporation

Volume Editors

Jan Zizka,
Mendel University in Brno, Czech Republic
E-mail: zizka.jan@gmail.com

Dhinaharan Nagamalai,
Wireilla Net Solutions, Australia
E-mail: dhinthia@yahoo.com

ISSN: 2231 - 5403
ISBN: 978-1-921987-51-9
DOI : 10.5121/csit.2016.60601 - 10.5121/csit.2016.60621

Typesetting: Camera-ready by author, data conversion by NnN Net Solutions Private Ltd., Chennai, India

Preface

The Sixth International Conference on Computer Science, Engineering and Information Technology (CCSEIT-2016) was held in Vienna, Austria, during May 21~22, 2016. The Third International Conference on Artificial Intelligence and Applications, The Third International Conference on Data Mining and Database (DMDB-2016), The Fifth International Conference on Mobile & Wireless Networks (MoWiN-2016), The Third International Conference on Computer Science and Information Technology (CoSIT-2016), The Second International Conference on Cryptography and Information Security (CRIS-2016), The Third International Conference on Signal and Image Processing (SIGL-2016), The Third International Conference on Bioinformatics and Bioscience (ICBB-2016) and The Ninth International Conference on Security and its Applications (CNSA-2016) were collocated with the CCSEIT-2016. The conferences attracted many local and international delegates, presenting a balanced mixture of intellect from the East and from the West.

The goal of this conference series is to bring together researchers and practitioners from academia and industry to focus on understanding computer science and information technology and to establish new collaborations in these areas. Authors are invited to contribute to the conference by submitting articles that illustrate research results, projects, survey work and industrial experiences describing significant advances in all areas of computer science and information technology.

The CCSEIT-2016, DMDB-2016, MoWiN-2016, CoSIT-2016, CRIS-2016, SIGL-2016, ICBB-2016, CNSA-2016 Committees rigorously invited submissions for many months from researchers, scientists, engineers, students and practitioners related to the relevant themes and tracks of the workshop. This effort guaranteed submissions from an unparalleled number of internationally recognized top-level researchers. All the submissions underwent a strenuous peer review process which comprised expert reviewers. These reviewers were selected from a talented pool of Technical Committee members and external reviewers on the basis of their expertise. The papers were then reviewed based on their contributions, technical content, originality and clarity. The entire process, which includes the submission, review and acceptance processes, was done electronically. All these efforts undertaken by the Organizing and Technical Committees led to an exciting, rich and a high quality technical conference program, which featured high-impact presentations for all attendees to enjoy, appreciate and expand their expertise in the latest developments in computer network and communications research.

In closing, CCSEIT-2016, DMDB-2016, MoWiN-2016, CoSIT-2016, CRIS-2016, SIGL-2016, ICBB-2016, CNSA-2016 brought together researchers, scientists, engineers, students and practitioners to exchange and share their experiences, new ideas and research results in all aspects of the main workshop themes and tracks, and to discuss the practical challenges encountered and the solutions adopted. The book is organized as a collection of papers from the CCSEIT-2016, DMDB-2016, MoWiN-2016, CoSIT-2016, CRIS-2016, SIGL-2016, ICBB-2016, CNSA-2016.

We would like to thank the General and Program Chairs, organization staff, the members of the Technical Program Committees and external reviewers for their excellent and tireless work. We sincerely wish that all attendees benefited scientifically from the conference and wish them every success in their research. It is the humble wish of the conference organizers that the professional dialogue among the researchers, scientists, engineers, students and educators continues beyond the event and that the friendships and collaborations forged will linger and prosper for many years to come.

Jan Zizka
Dhinaharan Nagamalai

Organization

General Chair

Jan Zizka	Mendel University in Brno, Czech Republic
Dhinaharan Nagamalai	Wireilla Net Solutions, Australia

Program Committee Members

A.K.M. Fazlul Haque	Daffodil International University, Bangladesh
Abd El-Aziz Ahmed	Cairo University, Egypt
Abdelhamid A.Mansor	University of Khartoum, Sudan
Abdelmounaim Abdali	Cadi Ayyad University, Morocco
Abdolreza Hatamlou	Islamic Azad University, Iran
Aiden B.Lee	University of La Jolla, USA
Albertus Joko Santoso	University of Atma Jaya Yogyakarta, Indonesia
Ali AL-zuky	Mustansiriyah University, Iraq
Ali Hussein	Alexandria University, Egypt
Aloizio	Aeronautic Institute of Technology, Brasil
Amol D Mali	University of Wisconsin, USA
Andino Maseleno	STMIK Pringsewu, Indonesia
Ankit Chaudhary	Truman State University, USA
Apai	Universiti Malaysia Perlis, Malaysia
Asmaa Shaker Ashoor	Babylon University, Iraq
Assem Abdel Hamied Moussa	ASDF/E commerce Manager, Egypt
Ayad Ghany	Erbil Polytechnic University, Iraqi Kurdistan
Ayad Salhieh	Australian College of Kuwait, Kuwait
Baghdad Atmani	University of Oran Ahmed Benbella, Algeria
Cherif Foudil	Biskra University, Algeria
Chin-Chih Chang	Chung Hua University, Taiwan
Daniel D. Dasig	Jr., Jose Rizal University, Philippines
Doina Bein	The Pennsylvania State University, USA
Emilio Jiménez Macías	University of La Rioja, Spain
Erritali Mohammed	Sultan Moulay Slimane University, Morocco
Farhad Soleimanian Gharehchopogh	Hacettepe University, Turkey
Fatih Korkmaz	Karatekin University, Turkey
Grigorios N.Beligiannis	University of Patras, Greece
Héldon José Oliveira Albuquerque	Integrated Faculties of Patos, Brazil
Hossein Jadidoleslamy	University of Zabol, Iran
Hou Cheng-I	Chung Hua University, Taiwan
Houcine Hassan	Univeridad Politecnica de Valencia, Spain
Isa Maleki	Islamic Azad University, Iran
Islam Atef	Alexandria University, Egypt
Israashaker Alani	Ministry of Science and Technology, Iraq
Jamal Zraqou	Isra University, Jordan
Jan Lindström	MariaDB Corporation, Finland

Technically Sponsored by

Networks & Communications Community (NCC)

Computer Science & Information Technology Community (CSITC)

Digital Signal & Image Processing Community (DSIPC)

Organized By

Academy & Industry Research Collaboration Center (AIRCC)

TABLE OF CONTENTS

The Sixth International Conference on Computer Science, Engineering and Information Technology (CCSEIT 2016)

The Third International Conference on Artificial Intelligence and Applications

The Third International Conference on Data Mininig and Database (DMDB 2016)

The Fifth International Conference on Mobile & Wireless Networks (MoWiN 2016)

The Third International Conference on Computer Science and Information Technology (CoSIT 2016)

The Second International Conference on Cryptography and Information Security (CRIS 2016)

The Third International Conference on Signal and Image Processing (SIGL 2016)

The Third International Conference on Bioinformatics and Bioscience (ICBB 2016)

The Ninth International Conference on Security and its Applications (CNSA 2016)

SMART AS A CRYPTOGRAPHIC PROCESSOR

Saroja Kanchi[1], Nozar Tabrizi[2] and Cody Hayden[3]

[1]Department of Computer Science, Kettering University, Flint, USA
skanchi@kettering.edu
[2]Department of Electrical and Computer Engineering, Flint, USA
ntabrizi@kettering.edu
[3]Department of Computer Science, Kettering University, Flint, USA
hayd7857@kettering.edu

ABSTRACT

SMaRT is a 16-bit 2.5-address RISC-type single-cycle processor, which was recently designed and successfully mapped into a FPGA chip in our ECE department. In this paper, we use SMaRT to run the well-known encryption algorithm, Data Encryption Standard. For information security purposes, encryption is a must in today's sophisticated and ever-increasing computer communications such as ATM machines and SIM cards. For comparison and evaluation purposes, we also map the same algorithm on the HC12, a same-size but CISC-type off-the-shelf microcontroller, Our results show that compared to HC12, SMaRT code is only 14% longer in terms of the static number of instructions but about 10 times faster in terms of the number of clock cycles, and 7% smaller in terms of code size. Our results also show that 2.5-address instructions, a SMaRT selling point, amount to 45% of the whole R-type instructions resulting in significant improvement in static number of instructions hence code size as well as performance. Additionally, we see that the SMaRT short-branch range is sufficiently wide in 90% of cases in the SMaRT code. Our results also reveal that the SMaRT novel concept of locality of reference in using the MSBs of the registers in non-subroutine branch instructions stays valid with a remarkable hit rate of 95%!

KEYWORDS

CISC and RISC comparison; Communication Security; Cryptography; Data Encryption Standard; Microprocessors; SMaRT

1. INTRODUCTION

Sixteen-bit microcontrollers are widely used in a variety of embedded systems such as power tools, medical instruments, toys, office products, automotive industry, remote controls and appliances [1]. Texas Instruments manufactures the popular family of MSP430 [2]. Microchip produces the well-known PIC24 MCUs and dsPIC® DSCs [3] . The S12XE family of automotive and industrial microcontrollers, as another example of 16-bit modern processors, is manufactured by NXP. For the list of 16-bit microcontrollers from NXP see [4].

In addition to industry, sixteen-bit microcomputers are commonly used in academia as well. There are numerous textbooks such as [5][6][7] based on 16-bit microcontrollers on the market.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 01–11, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60601

Additionally, 16-bit microcontroller-based education/training boards such as HCS12-based Dragon12Plus [8], and the PIC24-based Explorer 16 [9] are popular in academia.

Sixteen-bit microprocessors are not only a research topic [10][11][12][13], they are also used by researchers as a research tool. Tang et al use the Microchip PIC18F4520 to design an embedded controller for a portable fuel cell [14]. A 16-bit dsPIC is used in [15] for motion control of a mobile robot.

SMaRT is a Small Machine for Research and Teaching, which was recently designed and mapped into an Altera Cyclone II FPGA chip in our ECE department as reported in [16]. It is a 16-bit RISC-type single-cycle architecture with 16-bit long instructions. Unlike some 16-bit processors, SMaRT instruction-memory address-bus is 16 bits wide as well. This results in a better code density. Featuring the novel 2.5-address instructions, SMaRT can avoid data loss that inherently exists in 2-address machines. Additionally and as elaborated in [16], SMaRT's short branch instructions take advantage of the temporal locality of reference in accessing the upper or lower halves of the CPU's 16x16 orthogonal register file. This enables SMaRT to extend the range of the short branch instructions by a factor of 4. SMaRT is reviewed in Section 2.

In this paper we use SMaRT as a cryptographic processor. The advent of world-wide communication over the network makes cryptography essential to provide data transmission with security. Data encryption has been used for a long time. Security requirements may vary depending on the type of application and performance standards in terms of level of privacy, time, and power consumption leading to various encryption algorithms.

One of the most popular cryptographic algorithms is the Data Encryption Standard (DES) that converts plaintext to cipher text. The DES algorithm was standardized in 1977 by NIST. DES is the best symmetric cipher and is used in ATM machines and SIM cards. DES was replaced with Advanced Encryption Standard (AES) in 2000. It will be years before Advanced Encryption Standard (AES) can replace DES usage [17].

DES is based on a sequence of confusion and diffusion steps. The confusion step obscures the relationship between plaintext and cipher text and diffusion step ensures that a small change in plaintext causes significant changes in the cipher text. In this paper we map DES to the SMaRT processor.

An FPGA implementation of DES using pipelining, logic replication and register retiming is presented in [18]. A single-chip implementation of an iterative DES algorithm on a FPGA platform using 224 combinational logic blocks (CLBs) and 54 input/output blocks (IOBs) is presented in [19]. Patterson [20] presented a FPGA implementation of DES using Java API bit stream support for computing the key schedule entirely using software resulting in a throughput of 10 Gigabits per second. Another FPGA-based implementation of DES is presented in [21]. In this non-software-based design a 16-stage pipelined architecture is used to get the fastest DES implementation on a FPGA. In [22], Kaps et al use loop unrolling as well as pipelining to enhance their FPGA-based DES performance. An implementation of the DES algorithm using hardware loops and their variations can be seen in [23]. Standaert F.-X. et al present another FPGA-based implementation for DES and triple DES in [24]. They show that modern FPGAs provide sufficient resources to implement masked DES, hence improve security against power analysis attacks.

The rest of the paper is organized as follows: Overviews of SMaRT and Data Encryption Standard Algorithm (DES) are presented in Section 2. In Section 3 we map the above algorithm on SMaRT. We discuss our results in Section 4. Section 5 is the conclusion.

2. OVERVIEW

In this section, we first look at the SMaRT and then review the data encryption standard algorithm.

2.1. SMaRT

SMaRT has a 16x16 register file: R0 through R15. There are four different instruction formats in this machine, namely R, LSI, B and BL, as shown in Figure 1. Each SMaRT instruction is 16 bits wide (same as the data-bus width) with an exception of baleq and balne, which are 2 words long. Each one-word instruction and two-word instruction executes in one cycle and two cycles, respectively. The OpCode is always 3 bits wide and occupies bits 12 through 14 of each instruction.

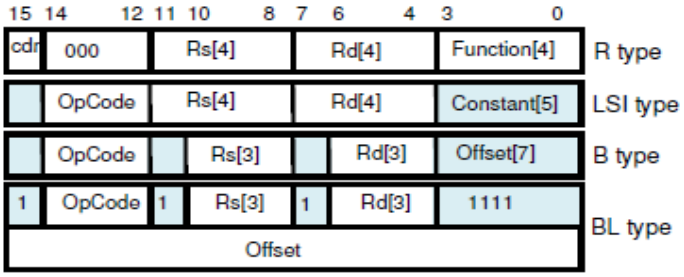


Figure 1. SMaRT instruction formats

The R-type instructions share the same OpCode; the Function field (bits 0 through 3) distinguishes between two such instructions. See Figure 1.

An R-type instruction may function as a 2-address or 2.5-address instruction based on the value of cdr bit, the MSB of the instruction. In a 2.5-address instruction, the operation result is stored in the register located right after the first operand register ruling out the data loss that exists in 2-address instructions. For example, while register R2 is overwritten hence lost in the following 2-address instruction

sub R2, R6

none of the source registers will be overwritten in the following 2.5-address instruction:

sub R2+, R6

Note that in the first instruction, R2 becomes R6 – R2; however, in the second instruction R3 becomes R6 – R2.

In branch instructions, Rs and Rd fields are only 3 bits wide; their MSBs are hidden and will be taken at run time from msbRs and msbRd, two flip-flops in the CPU. These flip-flops are updated by LSI- and R-type instructions. This way SMaRT may take advantage of the temporal locality of reference in accessing the two upper and lower halves of the register file. This means that, for

example, it is very likely that a branch instruction can use the lower half of the register file if the most recent LSI- or R-type instruction also uses the lower half. When this temporal locality of reference fails, the programmer may use *sff* instruction to explicitly set the MSB flip flops. SMaRT instruction summary is shown in Table 1.

Table 1. SMaRT instruction summary

Example	Meaning	Type	Comments
sub R3, R6	R3 <= R6 – R3	R	R3 is lost
sub+ R3, R6	R4 <= R6 – R3	R	R3 is not lost Set cdr = 1 for 2.5 address
slt+ R3 R6	If R6 < R3 R4 <= 1 Else R4 <=0	R	2.5 address R4 (destination) is located right after R3
addi R1 R3 7	R1 <= R3 + 7	LSI	Signed constant
lw R2, 9(R4)	R2<=mem(R4+9)	LSI	Signed offset
sw 9(R4) R2	mem(R4+9)<=R2	LSI	Signed offset
beq R3 R1, 7	If R3 = R1 then take next instruction from PC + 1 + 7 Else continue sequentially	B	Signed offset is 7 bites wide
bne R3 R1, 7	Same as above but now condition is R3 ≠ R1	B	So range is +-64
sff 1, 0	Set MSB FFs: msbRs <= 1 msbRd <= 0	B	See *
rtn	Take next instruction from memory location pointed to by R1	B	Use it to return from subroutine
baleq R3, R1 100A	If R3 = R1 then take next instruction from PC + 1 + 100A and place return address in R1 Else continue sequentially	BL	16-bit offset is 2nd word of instruction. See *
balne R3, R1 100A	Same as above but now condition is R3 ≠ R1	BL	Same as above

* In branch instructions, Rs and Rd fields are only 3 bits wide; their MSBs are hidden and will be taken at run time from msbRs and msbRd, the two MSB flip-flops in the CPU. These flip-flops are updated by LSI- and R-type instructions. This way SMaRT may take advantage of the temporal locality of reference in accessing the two upper and lower halves of the register file. This means that, for example, it is very likely that a branch instruction can use the lower half of the register file if the most recent LSI- or R-type instruction also uses the lower half. When this temporal locality of reference fails, the programmer may use *sff* instruction to explicitly set the MSB flip flops. sff is also explained above.

2.1. Data Encryption Standard Algorithm

In this section we present a brief overview of DES algorithm. More details can be found in [17].

The DES algorithm encrypts a 64-bit plaintext into a 64-bit cipher text using a 64-bit key. See Figure 2. It first employs an Initial Permutation (IP) on the plain text followed by 16 rounds of encryption and finally applies the final permutation (IP-1). As shown in the DES structure of Figure 2, the input to the first round is obtained after the initial permutation is applied to the 64-bit plaintext. The key schedule generates the key for each round of DES.

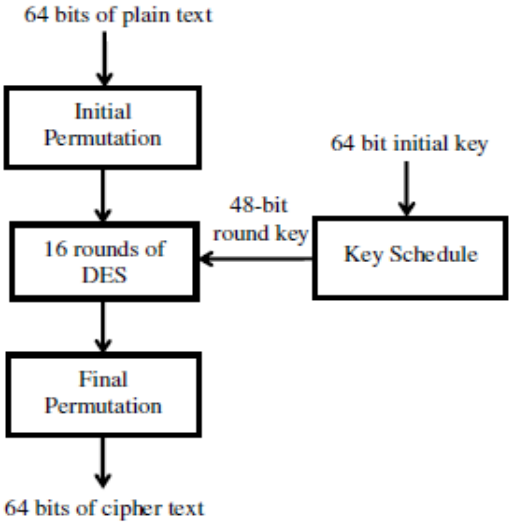


Figure 2. DES structure

The 64-bit input to each round i of DES can be viewed as consisting of the left half, L_{i-1}, and the right half, R_{i-1}. L_{i-1} is XORed with the result of the f-function which takes as input the 32-bit R_{i-1}, and a 48-bit round key k_i, to produce R_i, the right half output of round i. R_{i-1} then becomes the left-half output, L_i, for round i. That is,

$L_i = R_{i-1}$,

$R_i = L_{i-1} \oplus f(R_{i-1}, k_i)$

Round i is graphically depicted in Figure 3.

The f-function shown in Figure 4 contains four steps. First, is an Expansion step that expands the 32-bits $(b_i \;\; 1 \le i \le 32)$, into 48-bits, wherein the bit sequence $(b_i\ b_{i+1}\ b_{i+2}\ b_{i+3})$ is expanded into $(b_{(i-1) mod\ 32}\ b_{i\ mod\ 32}\ \ b_{(i+1)\ mod\ 32}\ \ b_{(i+2)\ mod\ 32}\ \ b_{(i+3)\ mod\ 32}\ \ b_{(i+4)\ mod\ 32})$, for each $i = 4k+1,\ 0 \le k \le$

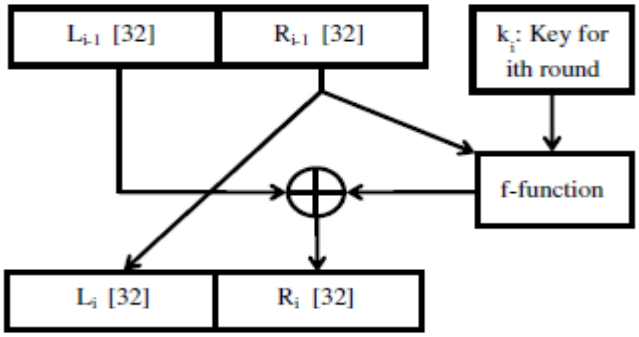


Figure 3. DES Round i

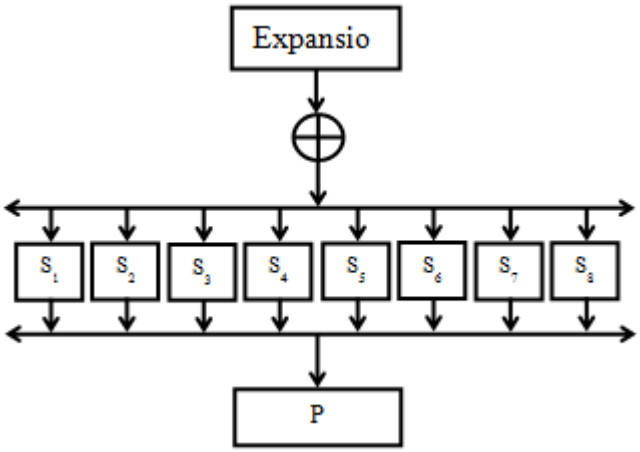


Figure 4. The f-function

The second step of the *f*-function is the XOR of the 48-bit output of Expansion with the 48-bit round key, k_i.

This is followed by a third step of S-box reduction from 48-bits to 32-bits. The S-box reduction splits the 48 bits into 8 sets of 6-bits from left to right. Then the eight S-boxes $S_i, 1 \leq i \leq 8$ are used to reduce the 6-bits to 4-bits. The first and last bits of the 6-bits are used as the row number of the S-box and the 4-bits are used as the column number. The first S-box, S1, is shown in Figure 5. The remaining S-boxes can be found in [15]. For example, the bit pattern 110011 will look up row number 11 (3) and column number 1001 (9) using S1 thus replacing the string 110011 with 4-bits 1011 (11).

Finally the fourth step is a permutation P [15] applied to the 32-bit output of S-box reduction.

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

Figure 5. A sample S-box (S_1)

The key schedule is depicted in Figure 6. The 56-bit key, k, and 8-bit parity constitute the 64-bit key in the initial step. Then the permutation PC-1 is applied on 56 non-parity bits to obtain 56-bit key for the first transformation. Each transformation for rounds 1 to 16 consist of rotations of each half of the key, followed a permutation PC- 2 which reduces the 56 bits to 48 bits.

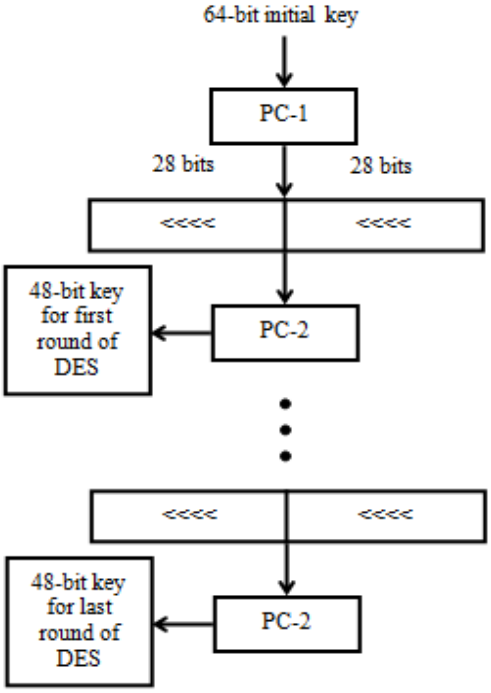


Figure 6. Key schedule for DES

3. ALGORITHM MAPPING

Due to its architecture, SMaRT processor immediately lends itself to several operations of the DES algorithm. Many steps in this algorithm are permutations including the Initial Permutation, PC-1, Final Permutation, PC-2, and the permutation P in the f-function in each round of DES. The pseudo code in Algorithm 1 shows the general steps used in all the permutations (all registers are just examples to help illustrate the algorithm):

R1 <= original text from memory location

R2 <= mask from memory location

R3 <= R1 AND R2

R4 <= number of times to rotate

R4 <= R3 rotated R4 times

R5 <= R4 OR R5

Repeat for each bit

Memory location <= R5

Algorithm 1. Permutation

An algorithm similar to Algorithm 1 is used in the Expansion function, the S-box lookup, and the left-shift-key transform. The difference in the Expansion function is that it masks multiple bits at a time before rotating and applying the OR operation. The Expansion function works by masking a group of six bits, following the general permutation steps illustrated in Algorithm 1, rotating the

mask four times to the right, and masking the next group of six bits. Because SMaRT has 16-bit registers, this group of six bits may be split between two registers, for example having two bits at the end of one register and four bits at the beginning of the next register.

The S-box lookup works by finding the correct address and loading the portion of memory, using that address, which has the lookup table for the S-boxes. The S-box lookup follows similar steps as the Algorithm 1 but it masks six bits at once and at the OR step another mask is combined with the rotated data to get the correct address. The first four S-boxes are in memory at location 01XX XXXX where the six X's are the six bits that are masked. The next four S-boxes are in memory at location 10XX XXXX, in the same way compared to the first four S-boxes. These six masked bits correspond to just one of the S-box numbers. For some examples of how this works, 0100 0000 holds the first number in the first four S-boxes, 0100 0010 holds the second number in the first four S-boxes, and 1000 0000 holds the first number in the last four S-boxes. In the first example, 0100 0000, the first four bits are 1110, the first number in the first S-box. The second four bits are 1111, the first number in the second S-box. This pattern repeats throughout the lookup table.

Finally the left-shift-key transform uses a technique similar to Algorithm 1. In this transform, every bit is rotated one to the left then the bits that need to be moved between registers are moved in the same way as the permutations move bits. For example, the last bit in the second register moves to the last bit in the first register, the last bit in the third register moves to the last bit in the second register, and the last bit in the fourth register moves to the last bit in the third register. In addition to these moves, the last bit in the first register is moved to be the twelfth bit in the second register and the twelfth bit in the second register is moved to be the eighth bit in the fourth register.

The XOR between the key and the data in the f function and the XOR after the f-function are implemented using the pseudo code given in Algorithm 2. (all registers are just examples to help illustrate the algorithm):

```
R1 <= first text from memory location
R2 <= second text from memory location
R2 <= R1 XOR R2
Memory location <= R2
```

Algorithm 2. XOR

The final steps in the DES algorithm consist of swapping the left half of the data with the right half of the data in memory. The pseudo code for these steps is illustrated in Algorithm 3.

```
R1 <= first text from memory location 1
R2 <= second text from memory location 2
R3 <= third text from memory location 3
R4 <= fourth text from memory location 4
Memory location 1 <= R3
Memory location 2 <= R4
Memory location 3 <= R1
Memory location 4 <= R2
```

Algorithm 3. Swap

4. RESULTS

Our studies in this research showed how remarkably SMaRT's features might help improve a SMaRT-based embedded system. We manually mapped the well-known Data Encryption Standard Algorithm on SMaRT. We also mapped the same algorithm on the HC12, an off-the-shelf microcontroller, but using a C compiler [25]. Our results showed that compared to the HC12, SMaRT code is only 14% longer in terms of the static number of instructions but some 10 times faster in terms of the number of clock cycles, and 7% smaller in terms of code size. 14% increase in the static code size is a very reasonable price for such a remarkable improvement. The significant difference between the numbers of clock cycles of the two processors is in part due to their architectural difference: SMaRT is a single-cycle RISC machine while HC12 is a CISC one with multi-cycle instructions. We then looked at the selling points of SMaRT and noticed some encouraging results: 278 R-type instructions out of 535, i.e. over 50%, are in the 2.5-address mode. Considering the total number of SMaRT instructions, this means that some 24% of the whole code takes advantage of this feature. We should have used some 278 more instructions if SMaRT had not had this novel feature. Our results also showed a hit-rate of 95% when the MSBs of the registers in non-subroutine branch instructions are taken from the most recently used LSI or R-type instruction. We also noticed that 11 SMaRT branch instructions out of 13 non-subroutine-call branches (in total) are short. In terms of dynamic number of branch instructions, this means that in 90% of times the SMaRT short branch instruction has a sufficient range.

5. CONCLUSION

In this paper we used SMaRT as a cryptographic processor. We mapped the Data Encryption Standard on SMaRT and showed that SMaRT's 2.5-address instructions comprise over 50% of the whole R-type instructions. This demonstrates how useful SMaRT's 2.5-address mode is. We also showed that for the register fields of SMaRT's non-subroutine branch instructions 3 bits are usually sufficient; the fourth bits are correctly taken from two flip-flops in the CPU at run time. These flip-flops are updated by LSI- and R-type instructions. We also showed that SMaRT's short branch instructions range is usually sufficient to reach the jump addresses. We additionally mapped DES on HC12 using the C language, and noticed that although SMaRT code is only 14% longer in terms of the static number of instructions, number of clock cycles for SMaRT is much lower than what HC12 needs.

REFERENCES

[1] https://www.futureelectronics.com/en/Microcontrollers/16-bit-microcontroller.aspx FUTURE ELECTRONICS 16bit Microcontrollers

[2] http://www.ti.com/lit/ug/slau144j/slau144j.pdf MSP430x2xx Family User's Guide

[3] http://ww1.microchip.com/downloads/en/DeviceDoc/70157F.pdf Microchip 16-bit MCU and DSC Programmer's Reference Manual

[4] http://www.nxp.com/products/microcontrollers-and-processors/more-processors/NXP 8/16 bit MCUs

[5] Cady Fredrick M. Software and Hardware Engineering: Assembly and C Programming for the Freescale HCS12 Microcontroller, 2nd Edition. ISBN: 0195308263

[6] Reese Robert B. et al. Microcontrollers: From Assembly Language to C Using the PIC24 Family, 2nd Edition. ISBN: 1305076559

[7] Almy Tom. Designing with Microcontrollers -- The 68HCS12. ISBN: 1463738501

[8] http://www.evbplus.com/download_hcs12/dragon12_plus_usb_9s12_manual.pdf Dragon12-Plus-USB Trainer for Freescale HCS12 microcontroller family; User's Manual for Rev. G board Revision 1.10

[9] http://ww1.microchip.com/downloads/en/DeviceDoc/Explorer%2016%20User%20Guide% 2051589a.pdf Microchip Explorer 16 Development Board User's Guide

[10] Gheorghe, A.-S. et al. 2010. Savage16 - 16-bit RISC architecture general purpose microprocessor, Intl Semiconductor Conf., CAS, Oct 2010, pp 521-524, Sinaia

[11] Morales-Velazquez et al. 2012. FPGA embedded single-cycle 16-bit microprocessor and tools, IEEE Intl Conf. on Reconfigurable Computing and FPGAs, pp 1-6, Cancun

[12] Muslim, S.M.S. et al. 2007. Design of an Algorithmic State Machine Controlled, Field Programmable Gate Array Based 16-bit Microprocessor, Intl Symposium on Integrated Circuits, ISIC '07, Sep. 2007, pp 434-436 Singapore

[13] Sakthikumaran S. et al. 2011. 16-Bit RISC Processor Design for Convolution Application. Intl Conf. on Recent Trends in Information Technology, ICRTIT, pp.394-397, Chennai

[14] Tang Tzu-Chiang et al. 2013. Embedded controller design for portable fuel cell, 9th Intl Conf. on Information, Communications and Signal Processing, ICICS 2013, pp 1-3, Tainan

[15] Suwannakom, A. 2014. Adaptive control performance of a mobile robot using hybrid of SLAM and fuzzy logic control in indoor environment, Intl Electrical Engineering Congress, March 2014, pp 1-4, Chonburi

[16] Tabrizi, N. 2016. SMaRT: Small Machine for Research and Teaching, International Journal of Electronics and Electrical Engineering (in press).

[17] Paar et al. 2010. Understanding Cryptography. Springer Publications

[18] Raed Bani-Hani1et al. 2014. "High-Throughput and Area-Efficient FPGA Implementations of Data Encryption Standard (DES)", Circuits and Systems, 2014, 5, 45-56.

[19] Wong K. et al. 1998. A Single-Chip FPGA Implementation of the Data Encryption Standard (DES) Algorithm, IEEE Global Telecommunications Conference on the Bridge to Global Integration, 8-12 November, 1998, Sydney, 827-832.

[20] Patterson C. 2000. High Performance DES Encryption in Virtex FPGAs using JBits. Proceedings of the 2000 IEEE Symposium on Field-Programmable Custom Computing Machines, Napa Valley, 17-19 April 2000, 113-121.

[21] McLoone M. et al. 2000. High-performance FPGA Implementation of DES. IEEE Workshop on Signal Processing Systems, Lafayette, 11-13 October 2000, 374-383.

[22] Kaps J.-P. et al. 1998. Fast DES Implementations for FPGAs and Its Application to a Universal Key-Search Machine. Selected Areas in Cryptography, Lecture Notes in Computer Science, 1556, 234-247

[23] Trimberger S. et al. 2000. A 12 Gbps DES Encryptor/Decryptor Core in an FPGA. Proceedings of the 2nd International Workshop on Cryptographic Hardware and Embedded Systems, Worcester, 17-18 August 2000, 156-163.

[24] Standaert F.-X., Rouvroy, G. and Quisquater, J.-J. (2006) FPGA Implementations of the DES and Triple-Des Masked Against Power Analysis Attacks. International Conference on Field Programmable Logic and Applications, 28-30 August 2006, Madrid, 1-4.

[25] ANSI-C/cC++ Compiler for HC12 V-5.0.41 Build 10203, Jul 23, 2010

INTENTIONAL BLANK

EFFECTS OF THE DIFFERENT MIGRATION PERIODS ON PARALLEL MULTI-SWARM PSO

Şaban Gülcü[1] and Halife Kodaz[2]

[1]Department of Computer Engineering,
Necmettin Erbakan University, Konya, Turkey
sgulcu@konya.edu.tr
[2]Department of Computer Engineering, Selcuk University, Konya, Turkey
hkodaz@selcuk.edu.tr

ABSTRACT

In recent years, there has been an increasing interest in parallel computing. In parallel computing, multiple computing resources are used simultaneously in solving a problem. There are multiple processors that will work concurrently and the program is divided into different tasks to be simultaneously solved. Recently, a considerable literature has grown up around the theme of metaheuristic algorithms. Particle swarm optimization (PSO) algorithm is a popular metaheuristic algorithm. The parallel comprehensive learning particle swarm optimization (PCLPSO) algorithm based on PSO has multiple swarms based on the master-slave paradigm and works cooperatively and concurrently. The migration period is an important parameter in PCLPSO and affects the efficiency of the algorithm. We used the well-known benchmark functions in the experiments and analysed the performance of PCLPSO using different migration periods.

KEYWORDS

Particle Swarm Optimization, Migration Period, Parallel Algorithm, Global Optimization

1. INTRODUCTION

In recent years, there has been an increasing interest in parallel computing. Software applications developed by using conventional methods run on a computer with limited resources as serial computing. Software executed by a processor on a computer consists of a collection of instructions. Each instruction is processed after another. An instruction is only processed at a time. But in parallel computing, multiple computing resources are used simultaneously in solving a problem. There are multiple processors that will work concurrently and the program is divided into different tasks to be simultaneously solved. Each task is divided into different instructions. The instructions are processed on different processors at the same time. Thus, performance increases and computer programs run in a shorter time. Parallel computing has been used in many different fields such as cloud computing [1], physics [2] and nanotechnology [3].

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 13–22, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60602

Recently, a considerable literature has grown up around the theme of metaheuristic algorithms. Particle swarm optimization (PSO) algorithm is developed by Kennedy and Eberhart in 1995 [4] is a popular metaheuristic algorithm. It is a population-based and stochastic optimization technique. It inspired from the social behaviours of bird flocks. Each individual in the population, called particle, represents a potential solution. In recent years, many algorithms based on PSO have been developed such as the comprehensive learning PSO (CLPSO) algorithm [5] and the parallel comprehensive learning particle swarm optimization (PCLPSO) algorithm [6]. In recent years, devising parallel models of algorithms has been a healthy field for developing more efficient optimization procedures [14-17]. Parallelism is an approach not only to reduce the resolution time but also to improve the quality of the provided solutions. In CLPSO, instead of using a particle's best information in the original PSO, all other particles' historical best information is used to update the particle's velocity. Further, the global best position of population in PSO is never used in CLPSO. With this strategy, CLPSO searches a larger area and the probability of finding global optimum is increased. The PCLPSO algorithm based on CLPSO has multiple swarms based on the master-slave paradigm and works cooperatively and concurrently. Through PCLPSO, the solution quality and the global search ability are improved. This article studies the effect of the different migration periods on PCLPSO algorithm.

This article has been organized in the following way: Section 2 is concerned with the methodologies used for this study. Section 3 presents the experimental results and the findings of the research. Finally, the article is concluded in Section 4.

2. MATERIALS & METHODS

2.1. PSO

Each particle in PSO represents a bird and offers a solution. Each particle has a fitness value calculated by fitness function. Particles have velocity information and position information updated during the optimization process. Each particle searches the food in the search area using the velocity and position information. PSO aims to find the global optimum or a solution close to the global optimum and therefore is launched with a random population. The particles update their velocity and position information by using Equations (1) and (2) respectively. To update the position of a particle, *pbest* of the particle and *gbest* of the whole population are used. *pbest* and *gbest* are repeatedly updated during the optimization process. Thus, the global optimum or a solution close to the global optimum is found at the end of the algorithm.

$$V_i^d = w * V_i^d + c_1 * rand1_i^d * (pbest_i^d - X_i^d) + c_2 * rand2_i^d * (gbest^d - X_i^d) \quad (1)$$

$$X_i^d = X_i^d + V_i^d \quad (2)$$

where V_i^d and X_i^d represent the velocity and the position of the dth dimension of the particle i. The constant w is called inertia weight plays the role to balance between the global search ability and local search ability [7]. c_1 and c_2 are the acceleration coefficients. *rand*1 and *rand*2 are the two random numbers between 0 and 1. They affect the stochastic nature of the algorithm [8]. $pbest_i$ is the best position of the particle i. *gbest* is the best position in the entire swarm. The inertia weight w is updated according to Equation (3) during the optimization process.

$$w(t) = w_{\max} - t * (w_{\max} - w_{\min}) / T \quad (3)$$

where w_{max} and w_{min} are the maximum and minimum inertia weights and usually set to 0.9 and 0.2 respectively [7]. t is the actual iteration number and T is the maximum number of iteration cycles.

2.2. CLPSO

CLPSO based on PSO was proposed by Liang, Qin, Suganthan and Baskar [5]. PSO has some deficiencies. For instance, if the *gbest* falls into a local minimum, the population can easily fall into this local minimum. For this reason, CLPSO doesn't use *gbest*. Another property of CLPSO is that a particle uses also the *pbest*s of all other particles. This method is called as the comprehensive learning approach. The velocity of a particle in CLPSO is updated using Equation (4).

$$V_i^d = w * V_i^d + c * rand_i^d * (pbest_{fi(d)}^d - X_i^d) \quad (4)$$

where $\boldsymbol{f}_i = [f_i(1), f_i(2), \ldots, f_i(D)]$ is a list of the random selected particles which can be any particles in the swarm including the particle i. They are determined by the Pc value, called as learning probability, in Equation (5). $pbest_{fi(d)}^d$ indicates the *pbest* value of the particle which is stored in the list $\boldsymbol{f}_i$ of the particle i for the dth dimension. How a particle selects the *pbest*s for each dimension is explained in [5].

$$V_i^d = w * V_i^d + c * rand_i^d * (pbest_{fi(d)}^d - X_i^d) \quad (5)$$

CLPSO uses a parameter m, called the refreshing gap. It is used to learn from good exemplars and to escape from local optima. The flowchart of the CLPSO algorithm is given in [5].

2.3. PCLPSO

Although PSO has many advantages, the main deficiency of PSO is the premature convergence [8]. PCLPSO handles to overcome this deficiency like many PSO variants. The PCLPSO algorithm based on CLPSO was proposed by Gülcü and Kodaz [6]. The solution quality is enhanced through multiswarm and cooperation properties. Also, computational efficiency is improved because PCLPSO runs parallel on a distributed environment.

A population is split into subpopulations. Each subpopulation represents a swarm and each swarm independently runs PCLPSO algorithm. Thus, they seek the search area. There are two types of swarms: master-swarm and slave swarm. In the cooperation technique, each swarm periodically shares its own global best position with other swarms. The parallelism property is that each swarm runs the algorithm on a different computer at the same time to achieve computational efficiency. The topology is shown in Figure 1. Each swarm runs cooperatively and synchronously the PCLPSO algorithm to find the global optimum. PCLPSO uses Jade middleware framework [9] to establish the parallelism. The cluster specifications are so: windows XP operating system, pentium i5 3.10 GHz, 2 GB memory, java se 1.7, Jade 4.2 and gigabit ethernet. The flowchart of the PCLPSO algorithm is given in [6].

In the communication topology, there isn't any directly communication between slave swarms as shown in Figure 1. Migration process occurs periodically after a certain number of cycles. Each swarm sends the own local best solution to the master in the PCLPSO's migration process. The master collects the local best solutions into a pool, called *ElitePool*. It chooses the best solution

from the *ElitePool*. This solution is sent to all slave swarms by the master. Thus, PCLPSO obtains better and more robust solutions.

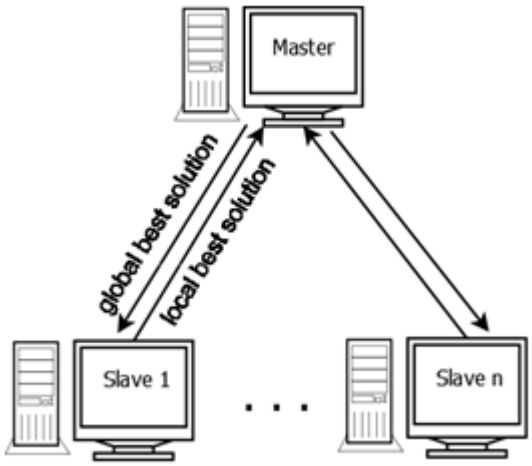


Figure 1. The communication topology [6]

3. EXPERIMENTAL RESULTS

The experiments performed in this section were designed to study the behaviour of PCLPSO by varying the migration period. The migration period is an important parameter in PCLPSO and affects the efficiency of the algorithm. This article studies the effect of the migration period on PCLPSO algorithm.

Two unimodal and two multimodal benchmark functions which are well known to the global optimization community and commonly used for the test of optimization algorithms are selected. The formulas of the four functions are given in next subsection. The properties of these functions are given in Table 1. The number of particles per swarm is 15. According to the dimensions of functions, the experiments are split into three groups. The properties of these groups are given in Table 2. The term FE in the table refers the maximum fitness evaluation.

The experiments are carried out on a cluster whose specifications are windows XP operating system, pentium i5 3.10 GHz, 2 GB memory, java se 1.7, Jade 4.2 and gigabit ethernet. The inertia weight w linearly decreases from 0.9 to 0.2 during the iterations, the acceleration coefficient c is equal to 1.49445 and the refreshing gap m is equal to five. 30 independent tests are carried out for each function. The results are given in next subsections.

Table 1. Type, Global Minimum, Function Value, Search and Initialization Ranges of the Benchmark Functions

f	Global Minimum x^*	Function Value $f(x^*)$	Search Range	Initialization Range
f_1	[0,0,…,0]	0	$[-100, 100]^D$	$[-100, 50]^D$
f_2	[1,1,…,1]	0	$[-2.048, 2.048]^D$	$[-2.048, 2.048]^D$
f_3	[0,0,…,0]	0	$[-32.768, 32.768]^D$	$[-32.768, 16]^D$
f_4	[0,0,…,0]	0	$[-600, 600]^D$	$[-600, 200]^D$

Table 2. Parameters used in experiments

Dimension	FE	Number of swarms	Number of particles
10	$3x10^4$	4	15
30	$2x10^5$	4	15
100	$3x10^5$	4	15

3.1. Functions

The functions used in the experiments are the following:

Sphere function:

$$f_1(x) = \sum_{i=1}^{D} x_i^2 \tag{6}$$

Rosenbrock function:

$$f_2(x) = \sum_{i=1}^{D-1} [100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2] \tag{7}$$

Ackley function:

$$f_3(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + e \tag{8}$$

Griewank function:

$$f_4(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \tag{9}$$

Functions f_1 and f_2 are unimodal. Unimodal functions have only one optimum and no local minima. Functions f_3 and f_4 are multimodal. Multimodal functions have only one optimum and many local minima. They are treated as a difficult class of benchmark functions by researchers because the number of local minima of the function grows exponentially as the number of its dimension increases [10-13].

3.2. Results of the 10-D problems

Table 3 presents the mean of the function values for 10-D problems according to the different migration periods. Table 4 presents the calculation time of the functions for 10-D problems. In [6], the importance of the migration period is emphasized: if the information is very often exchanged, then the solution quality may be better, but the computational efficiency deteriorates. If the migration interval is longer, the computational efficiency is better, but the solution quality may be worse. It is apparent from these tables that the computational efficiency is better when the migration interval is equal to 100 as expected. But the best values of functions f_1-f_4 are obtained when the migration intervals are equal to 11, 2, 6 and 1, respectively.

Table 3. The mean values for 10-D problems.

P	f_1	f_2	f_3	f_4
1	2.45e-03	7.20e+00	9.02e-02	**1.11e-01**
2	4.71e-03	**6.70e+00**	8.23e-02	1.38e-01
3	5.66e-03	7.51e+00	2.51e-01	1.49e-01
4	3.28e-03	7.03e+00	1.23e-01	1.43e-01
5	3.70e-03	7.68e+00	6.57e-02	1.28e-01
6	4.03e-03	7.94e+00	**6.49e-02**	1.29e-01
7	3.24e-03	7.32e+00	8.10e-02	1.36e-01
8	2.15e-03	7.17e+00	9.52e-02	1.38e-01
9	4.23e-03	7.90e+00	9.71e-02	1.40e-01
10	3.87e-03	8.98e+00	7.67e-02	1.25e-01
11	**1.97e-03**	7.17e+00	1.08e-01	1.28e-01
12	3.69e-03	7.78e+00	1.46e-01	1.43e-01
13	3.86e-03	8.26e+00	1.42e-01	1.03e-01
14	2.99e-03	7.16e+00	1.09e-01	1.24e-01
15	3.41e-03	8.49e+00	9.15e-02	1.18e-01
16	3.55e-03	8.79e+00	1.96e-01	1.27e-01
17	3.21e-03	7.47e+00	2.64e-01	1.32e-01
18	4.13e-03	8.16e+00	1.64e-01	1.37e-01
19	4.02e-03	7.08e+00	9.34e-02	1.51e-01
20	3.97e-03	6.84e+00	1.31e-01	1.29e-01
50	2.96e-03	7.92e+00	1.26e-01	1.30e-01
100	3.63e-03	6.90e+00	2.55e-01	1.25e-01

Table 4. The calculation time (ms) for 10-D problems

P	f_1	f_2	f_3	f_4
1	4463	4484	10359	15532
2	2230	2246	5184	7769
3	1484	1497	3450	5163
4	1122	1131	2600	3889
5	901	907	2081	3112
6	750	755	1732	2588
7	643	648	1482	2214
8	564	567	1303	1932
9	501	507	1158	1723
10	458	462	1051	1563
11	413	417	946	1406
12	377	380	864	1285
13	351	353	804	1187
14	322	326	736	1094
15	305	307	694	1031
16	289	290	657	975
17	271	272	614	911
18	252	254	573	846
19	243	244	551	815
20	234	236	530	784
50	101	102	220	319
100	**56**	**57**	**116**	**163**

Table 5. The mean values for 30-D problems.

P	f_1	f_2	f_3	f_4
1	**1.04e-09**	2.50e+01	1.49e-05	3.16e-06
2	1.63e-09	2.38e+01	1.07e-05	6.14e-07
3	3.17e-09	2.42e+01	1.48e-05	1.48e-06
4	2.09e-09	2.25e+01	1.39e-05	1.09e-06
5	1.12e-09	2.37e+01	1.77e-05	2.05e-06
6	2.40e-09	2.40e+01	1.35e-05	1.92e-05
7	4.05e-09	2.48e+01	1.08e-05	1.55e-06
8	1.37e-09	2.31e+01	1.51e-05	6.05e-07
9	2.15e-09	2.39e+01	1.31e-05	1.29e-05
10	1.51e-09	**2.17e+01**	1.65e-05	1.42e-06
11	1.52e-09	2.77e+01	**8.89e-06**	2.65e-06
12	1.93e-09	3.14e+01	1.35e-05	**3.12e-07**
13	1.32e-09	2.22e+01	1.20e-05	8.73e-07
14	2.33e-09	2.59e+01	1.01e-05	7.74e-07
15	4.27e-09	2.24e+01	1.07e-05	5.33e-07
16	1.85e-09	2.53e+01	1.60e-05	1.99e-06
17	1.78e-09	2.49e+01	1.40e-05	7.22e-07
18	2.12e-09	2.53e+01	1.54e-05	4.80e-06
19	3.17e-09	2.29e+01	1.37e-05	7.04e-07
20	1.95e-09	2.52e+01	1.72e-05	1.71e-06
50	2.63e-09	2.49e+01	1.73e-05	9.63e-06
100	3.21e-09	2.29e+01	1.24e-05	2.37e-06

Table 6. The calculation time (ms) for 30-D problems

P	f_1	f_2	f_3	f_4
1	501775	507009	1172359	1974659
2	250866	253531	586153	987428
3	167366	169169	390778	658059
4	125544	126884	293141	493516
5	100431	101484	234481	394644
6	83734	84672	195575	329072
7	71819	72603	167788	282494
8	62819	63466	146556	246678
9	55866	56447	130387	219428
10	50291	50831	117366	197528
11	45772	46262	106825	179744
12	41866	42291	97666	164303
13	38700	39109	90266	151909
14	35984	36359	83941	141228
15	33578	33928	78306	131734
16	31519	31891	73522	123447
17	29675	29981	69169	116419
18	28031	28309	65288	109822
19	26503	26784	61772	103956
20	25150	25500	58628	98703
50	10106	10216	23447	39425
100	**5709**	**5859**	**12519**	**20522**

Table 7. The mean values for 100-D problems.

P	f_1	f_2	f_3	f_4
1	7.04e-03	1.41e+02	3.46e-01	7.03e-03
2	1.95e-02	1.50e+02	1.03e+00	9.48e-03
3	1.69e-02	2.23e+02	2.27e-02	1.22e-02
4	2.05e-02	1.46e+02	1.80e-02	2.76e-02
5	9.95e-03	1.54e+02	**8.78e-03**	4.91e-02
6	1.63e-02	9.65e+01	1.84e-02	1.53e-02
7	8.94e-03	1.81e+02	1.74e-02	6.66e-03
8	3.65e-02	9.89e+01	5.83e-01	2.43e-02
9	1.67e-02	9.64e+01	4.70e-01	1.90e-02
10	2.15e-02	1.49e+02	1.03e+00	5.71e-03
11	**4.19e-03**	1.30e+02	1.91e-02	1.08e-02
12	1.34e-02	2.04e+02	1.41e-02	7.42e-03
13	1.16e-02	9.75e+01	1.29e-02	7.88e-03
14	6.72e-02	9.82e+01	1.09e+00	1.97e-01
15	5.70e-01	9.34e+01	1.56e-02	**3.93e-03**
16	1.31e-02	1.80e+02	2.03e-02	5.04e-03
17	4.63e-02	**9.10e+01**	2.94e-02	8.61e-03
18	3.27e-02	1.48e+02	1.03e+00	1.99e-02
19	1.32e-02	9.74e+01	2.56e-02	2.58e-02
20	1.68e-02	1.01e+02	1.40e-02	1.83e-02
50	3.02e-02	9.60e+01	1.78e-02	1.46e-02
100	9.58e-03	1.52e+02	7.94e-01	2.89e-02

Table 8. The calculation time (ms) for 100-D problems.

P	f_1	f_2	f_3	f_4
1	3865343	3907344	8432688	14837734
2	1934047	1952688	4217828	7418000
3	1288563	1301906	2809469	4943000
4	966735	976625	2108328	3708093
5	773703	782109	1686797	2966766
6	644532	651125	1405250	2471500
7	552656	558188	1204468	2118610
8	484078	488656	1054594	1855094
9	430563	434047	937250	1648172
10	387219	391531	845172	1484891
11	351922	356875	766891	1348719
12	322094	325265	702344	1235453
13	297469	300390	648907	1141359
14	276625	279359	603359	1060469
15	258031	260594	563062	989703
16	241766	244140	527125	926532
17	227922	230109	496844	873141
18	214859	216891	468234	823079
19	203953	206016	444921	781469
20	193954	195843	422985	742984
50	78093	78797	170172	297797
100	**39468**	**39844**	**85391**	**149562**

3.3. Results of the 30-D problems

Table 5 presents the mean of the function values for 30-D problems according to the different migration periods. The best mean values of functions f1-f4 are obtained when the migration periods are equal to 1, 10, 11 and 12, respectively. Table 6 presents the calculation time of the function values for 30-D problems.

3.4. Results of the 100-D problems

Table 7 presents the mean of the function values for 100-D problems according to the different migration periods. The best mean values of functions f1-f4 are obtained when the migration periods are equal to 11, 17, 5 and 15, respectively. Table 8 presents the calculation time of the functions for 100-D problems.

4. CONCLUSIONS

The purpose of the current study was to determine the effect of the migration period on PCLPSO algorithm. PCLPSO based on the master-slave paradigm has multiple swarms which work cooperatively and concurrently on distributed computers. Each swarm runs the algorithm independently. In the cooperation, the swarms exchange their own local best particle with each other in every migration process. Thus, the diversity of the solutions increases through the multiple swarms and cooperation. PCLPSO runs on a cluster. We used the well-known benchmark functions in the experiments. In the experiments, the performance of PCLPSO is analysed using different migration periods. This study has shown that the calculation time decreases when the migration interval is longer. We obtained better results on some functions when the migration period is around 10. The migration period should be tuned for different problems. Namely, it varies with regard to the difficulty of problems. As future work, we plan to investigate the number of particles to be exchanged between swarms on the performance of the PCLPSO algorithm.

ACKNOWLEDGEMENTS

This research was supported by Scientific Research Projects Office of Necmettin Erbakan University (Project No: 162518001-136).

REFERENCES

[1] M. Mezmaz, N. Melab, Y. Kessaci, Y.C. Lee, E.-G. Talbi, A.Y. Zomaya, D. Tuyttens, A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems, Journal of Parallel and Distributed Computing, 71 (2011) 1497-1508.

[2] Z. Guo, J. Mi, P. Grant, An implicit parallel multigrid computing scheme to solve coupled thermal-solute phase-field equations for dendrite evolution, Journal of Computational Physics, 231 (2012) 1781-1796.

[3] J. Pang, A.R. Lebeck, C. Dwyer, Modeling and simulation of a nanoscale optical computing system, Journal of Parallel and Distributed Computing, 74 (2014) 2470-2483.

[4] J. Kennedy, R. Eberhart, Particle swarm optimization, 1995 Ieee International Conference on Neural Networks Proceedings, Vols 1-6, (1995) 1942-1948.

[5] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, Ieee T Evolut Comput, 10 (2006) 281-295.

[6] Ş. Gülcü, H. Kodaz, A novel parallel multi-swarm algorithm based on comprehensive learning particle swarm optimization, Engineering Applications of Artificial Intelligence, 45 (2015) 33-45.

[7] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on, IEEE, 1998, pp. 69-73.

[8] F. Van Den Bergh, An analysis of particle swarm optimizers, in, University of Pretoria, 2006.

[9] F.L. Bellifemine, G. Caire, D. Greenwood, Developing multi-agent systems with JADE, John Wiley & Sons, 2007.

[10] X. Yao, Y. Liu, G. Lin, Evolutionary programming made faster, Evolutionary Computation, IEEE Transactions on, 3 (1999) 82-102.

[11] B.-Y. Qu, P.N. Suganthan, S. Das, A distance-based locally informed particle swarm model for multimodal optimization, Evolutionary Computation, IEEE Transactions on, 17 (2013) 387-402.

[12] X. Li, Niching without niching parameters: particle swarm optimization using a ring topology, Evolutionary Computation, IEEE Transactions on, 14 (2010) 150-169.

[13] S.C. Esquivel, C.A. Coello Coello, On the use of particle swarm optimization with multimodal functions, in: Evolutionary Computation, 2003. CEC'03. The 2003 Congress on, IEEE, 2003, pp. 1130-1136.

[14] E. Alba, Parallel metaheuristics: a new class of algorithms, John Wiley & Sons, 2005.

[15] G.-W. Zhang, Z.-H. Zhan, K.-J. Du, Y. Lin, W.-N. Chen, J.-J. Li, J. Zhang, Parallel particle swarm optimization using message passing interface, in: Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, Volume 1, Springer, 2015, pp. 55-64.

[16] M. Pedemonte, S. Nesmachnow, H. Cancela, A survey on parallel ant colony optimization, Applied Soft Computing, 11 (2011) 5181-5197.

[17] E B. Li, K. Wada, Communication latency tolerant parallel algorithm for particle swarm optimization, Parallel Computing, 37 (2011) 1-10.

APPLICATIONS OF THE ERLANG B AND C FORMULAS TO MODEL A NETWORK OF BANKING COMPUTER SYSTEMS – MOVING TOWARDS GREEN IT AND PERFORMANT BANKING

Florin-Catalin ENACHE and Adriana-Nicoleta TALPEANU

The Bucharest University of Economic Studies, Bucharest, Romania

ABSTRACT

This paper surveys the contributions and applications of queueing theory in the field of banking data networks. We start by highlighting the history of IT and banks and we continue by providing information regarding the main prudential regulations on the banking area as Basel Accords and green IT regulations, that on one side generate more computing needs and on the other side promote conscientious use of the existing IT systems.
Continuing with a background of the network technologies used in Economics, the focus will be on the queueing theory, describing and giving an overview of the most important queueing models used in economical informatics. While the queueing theory is characterized by its practical, intuitive and subtle attributes, the queueing models are described by a set of 3 factors: an input process, a service process and a physical configuration of the queue or the queueing discipline.
The Erlang B and C mathematical definitions of formulas for a specific number of s servers, at the λ arrival rate, and the average service time will be described, used and confirmed by computer simulations of real queues usually found in the banking computing systems.
The goal is to provide sufficient information to computer performance analysts who are interested in using the queueing theory to model a network of banking computer systems using the right simulation model applied in real-life scenarios, e.g. overcoming the negative impacts of the European banking regulations while moving towards green computing.

KEYWORDS

queueing theory, banking system, Erlang B, Erlang C, computer network, economical informatics, banking regulation, computer simulation, Basel.

1. INTRODUCTION

From the first recorded bank in the world, Taula de la Ciudad, which opened in Barcelona in 1401, to the current known banks, the services provided by banks developed considerably. The Bank Taula de la Ciudad was founded as a treasury resource for the Catalonian government. Even if the bank is on record as the first official bank in the world, the practice of banking has been traced back for several centuries. [12]

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 23–39, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60603

Many histories position the crucial historical development of a banking system to medieval and Renaissance Italy and particularly the affluent cities of Florence, Venice and Genoa. The Bardi and Peruzzi families dominated banking in 14th century Florence, establishing branches in many other parts of Europe. [7]

Perhaps the most famous Italian bank was the Medici bank, established by Giovanni Medici in 1397. [6] The oldest bank still in existence is Monte dei Paschi di Siena, headquartered in Siena, Italy, which has been operating continuously since 1472. (Boland, 2009).

The banking development spreads from northern Italy throughout the Holy Roman Empire, and to northern Europe in the 15th and 16th century. Another important point in time is the development of the important innovations that took place in the 17th century, in Amsterdam, during the Dutch Republic, and in the 18th century, in London. Of course, the development heavily continues with the innovations in telecommunications and computing, in the 20th century, when the size of the banks and the geographical coverage increase, due to the development of the operations' side of the banks. During the well-known financial crisis from 2007–2008, all the banks were affected, some of them more than others, causing a specific attention to the banking regulations in the upcoming years.

On the IT side, even if usually most observers prefer and are expected to discuss about what is coming and not about what happened, we would like to highlight at least the three most important events in the IT history. The first event may be considered the document "First Draft of a Report on the EDVAC" published by John Von Neumann end of June 1945, consisting of the first documented discussion of the stored program concept and the blueprint for computer architecture to this day [13]. It is also called "the technological basis for the worldwide computer industry" [1]. The second important event may be considered the giving birth of the Ethernet, by Bob Metcalfe, in 1973, at the Xerox Palo Alto Research Center (PARC). The third event is in 1989, when Tim Berners-Lee circulated "Information management: A proposal" at CERN in which he outlined a global hypertext system.

1.1 Banking and IT regulations

All types of banks, being part of the banking system, have to comply with the banking regulations. It can be distinguished three classes of banking regulations: economic regulation, prudential regulation and monetary regulation. What we will deepen further is the prudential or prevented regulation, which is designed "to ensure efficient allocation of resources, to minimize the risks assumed by banks and to ensure stability and financial soundness of individual banks and of the banking system as a whole". [3]

On the IT side, the regulation topic become more and more discussed during the recent years due to the huge development of the IT world.

1.2 Basel Banking Accords and other banking regulation

Nowadays, the most important international banking regulations are the Basel Accords. These accords are published in Basel, Switzerland, by the Bank for International Settlements BIS in Basel, which is the central body in charge to develop and standardize banking regulations.

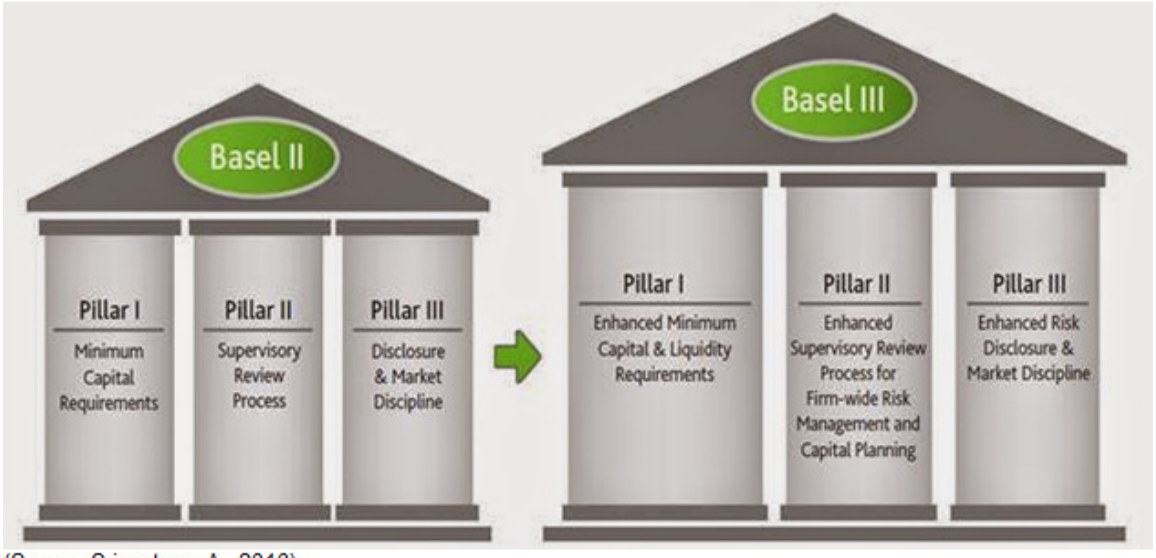
Basel I was the first accord adopted in 1988 and had the objective to improve the equity of internationally active banks by establishing a relation between equity and risk-weighted assets. Thus, Basel I proposed a standard methodology for calculating the equity and two solvability indicators to ensure compliance with the minimum coverage of risky assets (net exposure) through the bank capital. AT the end, it was concluded that the first indicator is enough for satisfying the minimum level of the solvability ratio 1 and the net exposure was calculated based on the credit risk, considering four risk categories (0%, 20%, 50% and 100%), applied according to the category of considered assets [11].

Basel II was adopted in 2004 and had the objective to cover various complains followed by Basel I. This accord contains changes to supervisory, regulatory and international cooperation between various authorities, his objectives being organised in three pillars: Pilar 1, Pilar 2 and Pilar 3. [3]
Pillar 1, also referred as "Minimum Capital Requirements", contains minimum capital requirements for credit risk, market risk and operational risk.

Pillar 2, also referred as "Supervisory Review Process", covers a qualitative approach about prudential requirements through the supervisory process. In addition to the risk defined in the Basel I, in Basel II the following risks are covered: the liquidity risk, residual risk, strategic risk, reputational risk, concentration risk and interest rate risk for exposures which are not in the trading book.

Pillar 3, also referred as "Market discipline", offers to the shareholders and the investors the possibility to monitor more effectively the bank management, because it requires to the banks to develop a set of detailed reporting requirements for the supervisory authority and for the public.
Basel III was issued as a result of the well-known global financial crisis, from 2007 and is improving several aspects of Basel II, a visual comparison being visible in the Figure 1. This accord requires from banks to have more equity of a superior quality, in order to be prepared to the future crisis, using Capital “Requirements Directive CRD IV” and “Capital Requirements Regulation CRR”. In addition, this accord defines a minimum leverage of 3% and two mandatory liquidity ratios: the rate of immediate liquidity and the long-term liquidity ratio. It is also enhancing the supervisory review process for firm-wide risk management and capital planning and the risk disclosure and the market discipline.

Worldwide, are currently existing also some other financial development institutions that took the role of supporting financial environment to adapt to a changing word. One of the largest ones is the International Finance Corporation (IFC), which hosts an informal group of banking regulators and banking associations, called Sustainable Banking Network. The group is currently designed to help regulatory authorities of emerging markets to develop green credit policies and environment and social risk management guidelines by sharing knowledge and technical resources. At the moment, the network has members from Bangladesh, Brazil, China, Indonesia, Lao PDR, Mongolia, Nigeria, Peru, Thailand and Vietnam.



(Source: Srivastava, A., 2016)

Fig. 1: Comparison between Basel II and Basel III

1.3 Green computing legislation

One of the first initiatives was taken place in USA, in 1992, and it was named Energy Star. This was a voluntary labelling program, created by the Environmental Protection Agency, having the purpose to promote energy efficiency in hardware components. In the recent years, the awareness about the necessity of a Green computing was increasing and therefore more directives appeared.

For example, the first WEEE Directive (Directive 2002/96/EC) entered into force in early 2003. WEEE stands for waste of electrical and electronical equipment such computers, TV-sets, fridges and cell phones. The directive has to goal to increase the recycling of WEEE and/or its reutilisation, by creating different collection schemes of WEEE, free of charge for population. [17]

The EU legislation (RoHS Directive 2002/95/EC) has to scope to restrict the use of hazardous substances in electrical and electronical equipment, referring specially to the heavy metals as mercury, lead, flame-retardants, cadmium, and why not, to find a cheaper and not that noxious substitutes. A newly revised directive by the European Commission became effective beginning of 2012. [17]

2. NETWORK TECHNOLOGIES USED IN ECONOMICS

With the global advancement seen in the last 20 years, especially with the increased volume, complexity, spread of exchanges in the economic and financial relations, all the computing systems, but especially the banking systems, had to adapt fast not only their banking regulations [16] [18] to the continuous changing world [12] but also their networking field, which had changed drastically over the time. Perhaps, the most fundamental change has been the rapid development of optical fiber technology. This has created limitless opportunities for new digital networks with greatly improved capabilities. The current broadband integrated service networks that provide integrated data, voice and video seem to have almost nothing in common with the

data networks of the last 20 years, but in fact, many of the underlying principles, mathematical and statistical laws are the same.

3. QUEUEING THEORY BACKGROUNDS – PROBABILITY, STOCHASTIC PROCESSES AND MATHEMATICS

Probability is a beautiful field of mathematics that is rich in its depth of deductive reasoning and in its diversity of applications. With its roots in the 17th century, probability started with simple counting arguments that were used to answer questions concerning possible outcomes of games of chance. Over the centuries, probability has been established as a key tool in a wide range of diverse fields like biology, chemistry, computer science, finance, medicine, physics, etc.

Probability served as the basis for deriving results to study stochastic processes. A stochastic process can be thought of as being a set of outcomes of a random experiment indexed by time. As an example, $X_n, n = 1, 2, 3, ...,$ could be the total number of tails obtained from the first n tosses of a fair coin in an experiment that continues for an indefinite period of time. The following set, $\{X_1, X_2, X_3, ...,\}$ represents a process to indicate that there is a relationship or dependency between the random variables X_n [5]. For continuous time processes, $X(t)$ is a stochastic process and the values of $X(t)$ and $X(t')$, for $t < t'$ have some kind of relationship or dependency.

A frequent application area for probability and stochastic processes is the queueing theory. The nomenclature used in queuing applications can be easily explained by the terms in Fig.2:

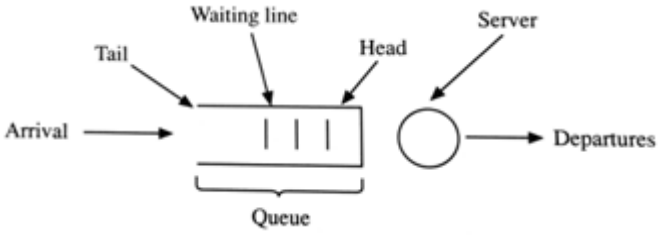


Fig.2. A single server queue

A single server queue consists of a server that processes customer requests, a waiting line or queue where customers wait before receiving service and an arrival stream of customers. For example, customers can arrive at the tail of the queue and are served on a first-come-first-served (FCFS or FIFO) basis [2]. In a computer model, the server could correspond to a hard drive that processes read or write requests, a CPU that processes customer requests or to a router serving a network of computers sending network requests. Typically, if the waiting room is finite, then any customer coming during the time when the waiting queue (the so called “buffer” in computer science terminology) is full is assumed to be lost to the system, just like if the customer never arrived.

The queuing theory is described by 3 fundamental characteristics:

- practical, as it has direct practical applications, for example, in modeling a network of computers in a banking or cloud environment

- intuitive, making it simpler to generate real-life models.

- subtle, making it intellectually interesting, because it uses probability to describe reality.

When speaking about a real-life model, it is common to have a multiple server queue, especially in modern CPU systems. Customers arrive at random, and try to find a server. If a server is available, they take it and hold it for a random amount of time, named onwards service time. If not, one possibility is that the customers that arrive when all servers are busy overflow and are lost (Erlang B formula). It is also possible that these customers wait in a queue (Erlang C formula).

A queueing model is defined by its:

- input process – a random process that defines the way in which the customers arrive, which are represented by the arrows in Fig.3.

- service process – a random process that defines the length of the service times needed by the arriving customers, which may be seen graphically by the height of the black bars above the arrows in Fig.3.

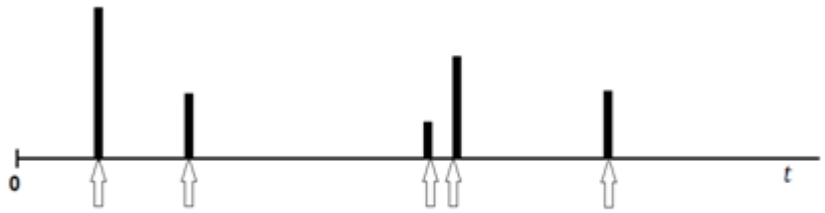


Fig.3. Random arrival with Random service times

- physical configuration or the queue discipline which defines what happens if a customer is blocked or waits in a queue. When the customers wait in a queue, more information is needed to describe the queue discipline, for example, the size of the queue, if a customer waits indefinitely or drops after a certain amount of time, the order in which a customer is served, e.g. last-come-first-served (LCFS) or in a random order etc.

4. THE ERLANG B FORMULA – PROBABILITY OF BLOCKING

For analysing this model, we are going to assume:

- *s* servers
- requests that arrive at a certain arrival rate λ
- an average service time τ

This queue will be intuitively analysed by using the classical *rate-up=rate-down* argument used in engineering, then the model will be limited and confirmed by using computer simulations.

The Fig. 3 below describes the system states for a queue with 3 services. Understanding the mechanism from Fig. 3 will help define a mathematical model and later develop the computer model simulation used to confirm the specific cases where this is available.

For this specific system we could be in state 0, 1, 2 or 3, which represents the number of customers present in the system - *N(t)*. We start at state 0, where no request is present in the system. Once the first customer arrives, the system jumps to state 1, it will stay there a while, until either that request completes or another request arrives. If another call arrives before the first one completes, then the system jumps to state 2, and stays there a while. Consequently, jumps down to 1, then it jumps up to state 2, and then maybe to state 3, and so one. If a customer arrives in state 3, then because all servers are busy, that would be a lost request. Practically every jump up corresponds to an arrival and all the jumps down correspond to a departure. *Rate up=rate down* means in this case that on the long term the number of the customers that arrive will equal to the number of customers that leave the system.

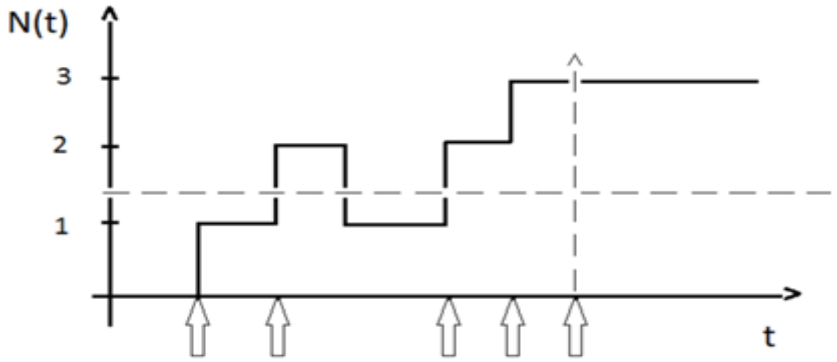


Fig.4. Random arrivals and random service times in a multi-server queue

The model will be analyzed by equating for each state the rate at which the system jumps up from that state to the rate the system jumps down from the state above it. If we look at the Fig.4., the dotted line shows that the number of jumps up and the number of jumps down will differ by at most 1, which in limit does not matter. That means that by dividing the number of jumps up by the total amount of time we get the rate up, and by dividing the number of jumps down by the total amount of time results the rate down. The rate up from $P_0 = \lambda P_0 = \frac{1}{\tau} P_1$ =rate down from P_1.

By the same argument, λP_1 would be the rate up from state 1, and $\frac{2}{\tau} P_2$ is the rate down from state 2, since from state 2 there are 2 chances for a completed request. Going further with this argument, the following set of equations is developed:

$$\lambda P_0 = \frac{1}{\tau} P_1$$

$$\lambda P_1 = \frac{2}{\tau} P_2$$

$$\lambda P_2 = \frac{3}{\tau} P_3 \qquad (1)$$

M

$$\lambda P_{s-1} = \frac{s}{\tau} P_s$$

The problem is reduced to solving this set of equations and finding P_s - the probability that all servers are busy. By normalizing and using the notation $\lambda\tau = a$, it is hints that the solution would depend on the number of servers and on the product $\lambda\tau = a$, called offered load, and not by the individual values of either λ or τ.

$$P_1 = \lambda\tau P_0 = aP_0$$

$$P_2 = \lambda\tau P_1 = \frac{a^2}{2} P_0$$

$$P_3 = \lambda\tau P_2 = \frac{a^3}{3!} P_0 \quad \Rightarrow P_j = \frac{a^j}{j!} P_0, j = 1,2,...,s \qquad (2)$$

M

$$P_s = \lambda\tau P_{s-1} = \frac{a^s}{s!} P_0$$

The set of equations is completed by the fact that:

$P_0 + P_1 + ... + P_s = 1 \Rightarrow P_0 + \frac{a^1}{1!} P_0 + \frac{a^2}{2!} P_0 + ... + \frac{a^s}{s!} P_0 = 1$, therefore,

$P_0 = \frac{1}{1 + \frac{a^1}{1!} + \frac{a^2}{2!} + ... + \frac{a^s}{s!}}$, resulting that $P_s = \frac{\frac{a^s}{s!}}{1 + \frac{a^1}{1!} + \frac{a^2}{2!} + ... + \frac{a^s}{s!}} = E_{1,s}(a) = B(s,a)$ (3),

also named the Erlang B formula, giving the probability of blocking, which is the probability that all servers are busy.

At this moment, it is quite easy to calculate how many servers would be needed in order to have, let's say less than 1% dropped requests. This is important on both financial and quality aspects. If too many servers are provided, then the service will be good, but the entire system will be more expensive than necessary. Looking at Fig. 5, we would draw the line corresponding to 0.01, estimate the offered load, and then search for the first curve that would lie below point of intersection. This points out the practical application of the Erlang B formula that helps engineers to calculate the number of servers needed for given values of the offered load and percent of lost requests.

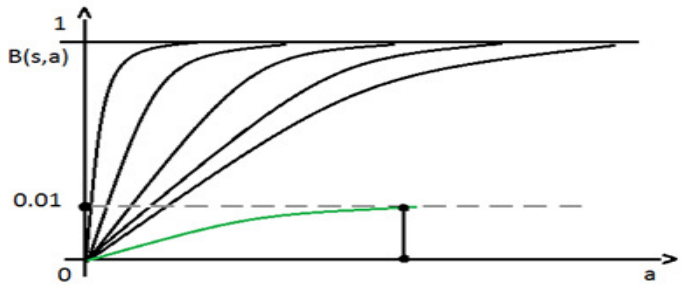


Fig.5. Erlang B graph

5. SUBTLETIES OF THE ERLANG B FORMULA – COMPUTER SIMULATIONS

Looking at Fig.6 we can observe the following arrival processes that have the same rate of arrival λ:

- constant arrival process – green
- random arrival process – black
- in between arrival process – red

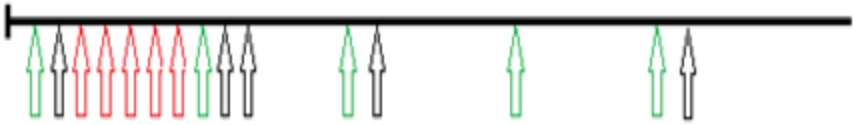


Fig.6. Different types of arrival processes

The question is which type of arrival process is described by the previous rationale. The rationale could be always right, never or just sometimes (Poisson exponential arrivals). Also this could be extended to the service process. To easily demonstrate that, a computer simulation will be developed.

The physical interpretation of B(s,a)=20% is one of the subtleties of the Erlang B formula. Is it that 20% of all requests are going to be lost? Another possibility is that 20% of the time all servers are busy. Further, is another question arises: what exactly is P_2 - the probability of being in state 2? Is that the fraction of arrivals who finds the system in state 2, or the fraction of time the system is in state 2?

The following computer code written in BASIC will calculate the fraction of time the system is in the blocking state (a ratio of times) and the fraction of customers who arrive during the blocking state (ration of integers – number of customers that overflow divided by the total number of customers who arrive). So the left and the right sides of the (1) equations are measured in complete different ways. The computer code assumes the arrival rate to be 4, the average service times to be 2.4 resulting an offered load of 4x2.4=9.6 erlangs, and the number of servers to be 10. The simulation will run for 100000 arriving requests, and the types of arrival and service time processes will be considered as follows:

- Poisson arrivals, exponential service times
- Poisson arrivals, constant service times
- Constant inter-arrival times, exponential service times
- Constant inter-arrival times, constant service times

```
100 DIM C(50) (50 is max number of servers)
110 INPUT S,NSTOP (S,NSTOP = number of servers, customers to be
    simulated)
120 FOR D=1 TO NSTOP
130 IA= (IA = interarrival time)
140 A=A+IA (A = arrival time)
150 J=0
160 J=J+1 (J = index of server being probed)
170 IF J=S+1 THEN K=K+1 (K = number of customers that are blocked)
180 IF J=S+1 THEN 270
190 IF A<C(J) THEN 160 (C(J) = completion time for server J)
200 X= (X = service time)
210 C(J)=A+X
220 M=C(1) (M = shortest server-completion time)
230 FOR I=2 TO S
240 IF C(I)<M THEN M=C(I)
250 NEXT I
260 IF M>A THEN AB=AB+M-A (AB = cumulative time during which all
     servers busy)
270 NEXT D
280 PRINT K/NSTOP,AB/A (fraction of customers blocked, fraction of time
    all servers are simultaneously busy)
```

6. SIMULATION INPUTS AND RESULTS FOR ERLANG B

Table 1. Erlang B – Simulation inputs and outputs

	IA[10]	X	K/NSTOP	AB/A	B(s,a)	
1	-(1/4)*LOG(1-RND)	-2.4*LOG(1-RND)	0.19652	0.1958451	19.6% [16]	
2	-(1/4)*LOG(1-RND)	2.4	0.19652	0.1958451		
3	1/4	-2.4*LOG(1-RND)	0.13366	.2360997	N/A	N/A
4	1/4	2.4	0	0.6009062	N/A	N/A

As seen in Table 1, the variability in the service times does not affect the answer, but the variability in the arrival process does.

No matter what distribution function we use to describe service times, the answer is unaffected. From a practical point of view, something that is difficult to measure, does not have to be measured because the final answer does not depend on it. We can completely disregard the statistical characteristics of the service times, once this model is understood. That is why these formulas are so robust and safe to use.

7. THE ERLANG C FORMULA – BLOCKED CUSTOMERS DELAYED

By extending the heuristic conservation-of-flow to include the case in which all customers who find all servers busy wait until they are served, and by following the same intuitive approach, rate up=rate down, we get the following set of formulas:

$$\lambda P_0 = \frac{1}{\tau} P_1$$

$$\lambda P_1 = \frac{2}{\tau} P_2$$

$$\lambda P_2 = \frac{3}{\tau} P_3 \quad \Rightarrow P_j = \frac{a^j}{j!} P_0, j = 0,1,...,s-1 \qquad (4)$$

M

$$\lambda P_{s-1} = \frac{s}{\tau} P_s$$

Further, when looking at the first set of formulas it was concluded that the rate down from state 2 is $\frac{2}{\tau}$, but at the point where all servers are busy, the aggregate service completion rate would be constant and equal to $\frac{s}{\tau}$ because all servers are busy, and only *s* customers are served. This leads to the following set of equations:

$$\lambda P_s = \frac{s}{\tau} P_{s+1}$$

$$\lambda P_{s+1} = \frac{s}{\tau} P_{s+2}$$

$$\lambda P_{s+2} = \frac{s}{\tau} P_{s+3} \quad (5) \quad \Rightarrow \quad P_{s+1} = \frac{a}{s} P_s = \frac{a^{s+1}}{s! s^1}, \quad P_{s+2} = \frac{a^{s+2}}{s! s^2}$$

M

By combining equations (4) and (5), the following formula is deducted for this model:

$$P_j = \begin{cases} \frac{a^j}{j!} P_0, (j = 1,2,...,s-1) \\ \frac{a^j}{s! s^{j-s}}, (j = s, s+1,...) \end{cases} \qquad (6)$$

By normalization (requirement that all probabilities add up to 1) we get:

$$P_0 + P_1 + ... + P_s + P_{s+1} + ... = 1 \Rightarrow P_0 (1 + \frac{a^1}{1!} + \frac{a^2}{2!} + ... + \underbrace{\frac{a^{s-1}}{(s-1)!} + \frac{a^s}{s!} + \frac{a^{s+1}}{s! s^1} + \frac{a^{s+2}}{s! s^2} + ...}_{\frac{a^s}{s!}(1+\left(\frac{a}{s}\right)^1+\left(\frac{a}{s}\right)^2+...)}) = 1 \qquad (7)$$

It is obvious that the right part of the above equation is an infinite geometric series and the formula only makes sense if the series converges. By mathematical reasoning, the series

converges to $\frac{1}{1-\left(\frac{a}{s}\right)}$; therefore, the formula makes sense if and only if $a<s$. We conclude that

$$P_0 = \frac{1}{\sum_{k=0}^{s-1}\frac{a^k}{k!} + \frac{a^s}{s!\left(1-\frac{a}{s}\right)}} \quad (8), \qquad \text{if } \frac{a}{s} < 1.$$

8. SUBTLETIES OF THE ERLANG C FORMULA

Since $\frac{a}{s}$ must be less than 1, then a has to be less than s, it results that $\lambda\tau < s$, therefore $\lambda < \frac{s}{\tau}$. This means that the arrival rate must be less than the maximum average completion rate, otherwise the queue is going to grow to infinity, being impossible to find an equilibrium state.
The Erlang B formula calculates the probability of the system to be in the blocking state, which is equal to the fraction of customers who find the blocking state and therefore are lost. In the case of the Erlang C formula, the fraction of time when the system finds itself in the blocking state is calculated, but in this case, the blocking state (probability of queueing) means not only state s, but also state s+1, s+2, etc., while limiting this rationale to Poisson input.

$$C(s,a) = P_s + P_{s+1} + P_{s+2} + ... = E(2,a) \text{ (by definition)}$$

$$C(s,a) = P_s + P_{s+1} + P_{s+2} + ... = P_0 \frac{\frac{a^s}{s!}}{a-\left(\frac{a}{s}\right)} = \frac{\frac{a^s}{s!\left(1-\frac{a}{s}\right)}}{\sum_{k=0}^{s-1}\frac{a^k}{k!} + \frac{a^s}{s!\left(1-\frac{a}{s}\right)}} = E_{2,s}(a) = \frac{sB(s,a)}{s-a(1-B(s,a)}$$

(9), giving the fraction of customers who have to wait in the queue in a model in which all blocked customers wait as long as necessary to be served.

Getting back to the *rate-up=rate-down* assumption, the following is true:

$$\lambda P_j = \mu_{j+1} P_{j+1}, \text{ where } \mu_k = \begin{cases} \frac{k}{\tau}, (k < s) \\ \frac{s}{\tau}, (k \geq s) \end{cases} \quad \textbf{(10)}$$

When considering the rate-up=rate-down argument, how long the system is in any particular state was not taken into consideration, but only that the system is in that state. The rate at which the system goes up from state j depends only on j - the number of customers present - , but it does not depend on the past history of the system, except that that the past history produces this current state. Likewise, if the system is in state j+1, the rate at which the system goes down (rate at which

customers leave the system) depends only on how many customers are currently present and not on the past history of the system. In other words, it does not depend on when the last arrival occurred, it depends on how much service remains for each of the customers who are being served. By using the *rate-up=rate-down* argument, the past history is neglected, and this implies that the underlying variables are exponential, because of the Markov property [9], which states that the only thing that affects the future evolution of a random variable that is exponential is its basic parameters but not how long it has been in progress. This is also an intuitive interpretation as to why a sufficient condition to this rationale is that all the underlying variables are exponential. One exception to this rule was analyzed earlier under the Erlang B formula when the blocked customers are cleared. The requirement for this is that we have Poisson input and exponential service times, where for Erlang B having Poisson arrivals but not necessarily exponential service times was enough.

In Fig.4, for the Erlang B as the offered load increases, the probability of blocking increases and is asymptotic to 1, getting a sequence of curves. For an increasing number of servers, the curve for a larger number of servers *s* lies below the curve for a smaller number of servers, because the larger number of servers will reduce the probability of blocking for the same value of *a* – offered load.

By plotting the Erlang C values, the result is similar, but not exactly the same, because now we have the condition that *a* must be less than *s* ($a < s$). For 1 server, when *a* is equal to 1 erlang, the curve would not be asymptotic to 1, but it will reach 1, because when the offered load is 1 or more, it means that the infinite series does not converge and the probability of waiting in a queue is 100%. Redoing the before mentioned thinking, we get the following graph:

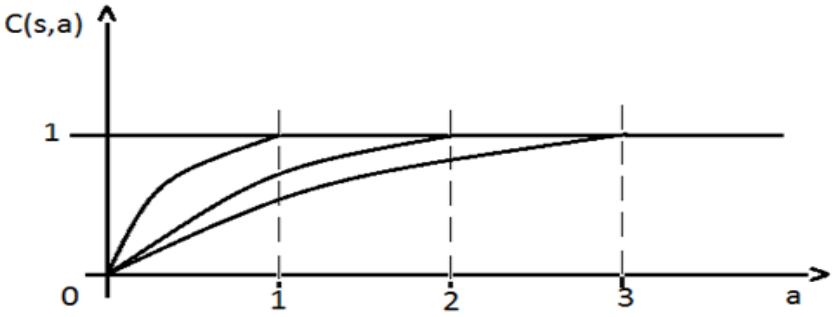


Fig.7. Erlang C graph

The model used in the Erlang C simulation is called M/M/s by the conventional queueing theory notation. The general model introduced by Kendall [19] in 1953 can be summarized as follows: *a/b/c*, where:

- *a* indicates the arrival process, where M represents the memoryless Markov property – Poisson input
- *b* indicates the service process, where M represents Markov memoryless property – exponential service times
- *c* is the number of servers, and it is implicitly that there is an infinite queueing capacity

The Erlang B model is by the same notation M/G/s/s, having Poisson input, general service times, *s* servers and the *s* capacity of the system (no waiting positions). Further details on the M/G/1 [9] model, including a simulation can be found in the reference [4].

The following computer code written in BASIC will simulate the same problem as with the Erlang B case, with the only difference that the blocked customers are now allowed to wait in a queue. The simulation will calculate what the customers see, more exactly the fraction of customers who have to wait in the queue, which is equal to the fraction of time the system is blocked, according to the PASTA Theorem [14]. The computer code assumes the arrival rate to be 4, the average service times to be 2.4 resulting an offered load of 4x2.4=9.6 erlangs, and the number of servers to be 10. The simulation will be run for 100000 arriving requests, and the types of arrival and service time processes will be considered as follows:

- Poisson arrivals, exponential service times. (M/M/s)
- Poisson arrivals, constant service times. (M/D/s)
- Constant inter-arrival times, exponential service times. (D/M/s)
- Constant inter-arrival times, constant service times. (D/D/s)

```
100 DIM C(50)
110 INPUT S, NSTOP
120 FOR D = 1 TO NSTOP
    130 IA = -(1 / 4) * LOG(1 - RND)
    140 A = A + IA
    180 M = C(1): z = 1
    190 FOR I = 2 TO S
        200 IF C(I) < M THEN M = C(I): z = I
    210 NEXT I
    220 X = -2.4 * LOG(1 - RND)
    221 sx = sx + X
    230 IF A > C(z) THEN C(z) = A + X ELSE C(z) = C(z) + X: w = C(z) -
  A: K = K + 1
    231 sw = sw + w
250 NEXT D
PRINT sx / A / 10, sw / A, K / NSTOP,SW/NSTOP
```

9. SIMULATION INPUTS AND RESULTS FOR ERLANG C

Table 2. Erlang C – Simulation inputs and outputs

s=10	1		2		3		4	
$\lambda = 4$	Theory	simulation	Theory	simulation	Theory	simulation	Theory	simulation
$\rho = \frac{a}{s}$	96%	0.9650543	96%	0.9704734	96%	0.9497337	96%	0.9600935
$E(W) = \frac{C(s,a)}{(1-\rho)s}\tau$ [8]	6.45762	6.469176	N/A	5.999352	N/A	4.222098	0	0
$P(W > 0) = C(s,a)$	0.859046	0.85792	N/A	0.8586	N/A	0.7386	0	0

10. CONCLUSIONS

Considering the facts presented in the paper, the environment, the IT and the banking systems should not be seen as single independent entities and the strong existing interlinkage between all of them should be considered. Currently, are already existing few regulations that are connecting the environment with the IT and the banking systems and some initiatives are planned for the upcoming years, but the maturity is not that high.

By using the Erlang formulas, we are able to use the above stated probabilities to represent both the point of view of an outside observer simply by looking at a system over the total time and calculating these probabilities as a fraction of time, and the point of view of the arriving customers who see the system only at the instance at which they arrive, meaning finding the system in blocking state and therefore waiting in a queue.

For the Erlang B formula, the variability in the service times does not affect the answer, but the variability in the arrival process does. No matter what distribution function we use to describe service times, the answer is unaffected. From a practical point of view, something that is difficult to measure, does not have to be measured because the final answer does not depend on it. We can completely disregard the statistical characteristics of the service times, once this model is understood. This is the reason why these formulas are so robust and safe to use.

Based on all information presented in this paper, we can conclude that computer simulation is an important tool for the analysis of queues whose service times have any arbitrary specified distribution. In addition, the theoretical results for the special case of Poisson arrivals and exponential service times are extremely important because they can be used to check the logic and accuracy of the simulation, before extending it to more complex situations.

Moreover, such a simulation gives insight on how such a queue would behave as a result of different arrival processes and service times. Further, we consider that it offers a methodology for looking into more complicated cases not only like getting input times from a network of banking systems trying to implement a new set of banking regulations where a mathematical approach cannot help, but also in other complex areas.

Another important point of the paper is to have mandatory environmental enriched IT and banking regulations applicable worldwide, not only to some regions or countries as it is happening today. In this way, the regulators will be challenged to not think in silos anymore, on a short term, and to adopt a broader view applicable for a long term period, because everything is connected and the environmental impacts are affecting the entire world.

REFERENCES

[1] Campbell-Kelly, M., & Aspray, I. (2004). Computer: A History Of The Information Machine (Sloan Technology). Westview Press

[2] Cooper, R. B. (1981). Introduction to Queueing Theory, Second Edition. New York: North Holland, New York.

[3] Dardac, N., Moinescu, B. (2007). Politici monetare şi tehnici bancare. Course notes.

[4] Enache, F.C. (2014). Stochastic Processes and Queueing Theory for Cloud Computer Performance Analysis, In: Conference Proceedings of the 14th International Conference on Informatics in Economy, pp13-19.

[5] Ghahramani, S. (2005). Fundamentals of Probability with Stochastic Processes, Third Edition. Upper Saddle River, Pearson Prentice Hall, New Jersey.

[6] Goldthwaite, R. A. (1995). Banks, Places and Entrepreneurs in Renaissance Florence: 492 (Variorum Collected Studies), Variorum.

[7] Hoggson, N. F. (1926). Banking Through the Ages. Dodd, Mead & Company.

[8] Lakatos, L. (2008). A note on the Pollaczek-Khinchin Formula. In: Annales Univ. Sci. Budapest, Sect. Comp. 29, pp. 83-91.

[9] Sigman, K.. (2011). Exact Simulation of the stationary distribution of the FIFO M/G/c Queue. J. Appl. Spec. Vol. 48A, pp. 209-213, <http://www.columbia.edu/~ks20/papers/QUESTA-KS-Exact.pdf>. [January 20, 2015].

[10] Sigman ,K.. (2010). Inverse Transform Method, <http://www.columbia.edu/~ks20/4404-Sigman/4404-Notes-ITM.pdf>. [January 15, 2015].

[11] Tarullo, D. K.. (2008). Banking on Basel: The Future of International Financial Regulation. Peterson Institute for International Economics, U.S.A.

[12] Van Dillen, J.G. (1964). History of the principal public banks.Frank Cass& CO LTD.

[13] Von Neumann, J. (1945). First Draft of a Report on the EDVAC. Contract No. W-670-ORD-4926 between the United States Army Ordnance Department and the University of Pennsylvania.

[14] Wolff, R. W. (1981). Poisson arrivals see time averages, <http://www2.isye.gatech.edu/~spyros/courses/IE7201/Fall-13/PASTA-proof.pdf>. [August 14, 2015].

[15] ***, (2015). Bank for International Settlements- Basel Committee on Banking Supervision, <http://www.bis.org/bcbs/>. [August 10, 2015].

[16] ***, (2015). Erlang B Calculator, < http://home.earthlink.net/~malcolmhamer/Erlang-B.xls>. [August 14, 2015].

[17] ***, (2010). The European Environment - State and Outlook 2010. European Environmental Agency.

[18] ***, (2015). European Banking Authority, <www.eba.europa.eu>. [September 02, 2015].

[19] ***, (2015). Kendall's notations, <https://www.andrewferrier.com/oldpages/queueing_theory/Andy/kendall.html>. [August 10, 2015].

ACKNOWLEDGEMENTS

The authors would like to thank their families for support!

AUTHORS

Florin-Catalin ENACHE graduated from the Faculty of Cybernetics, Statistics and Economic Informatics of the Academy of Economic Studies in 2008. Starting 2010 he holds a MASTER degree in the field of Economic Informatics, in the area of "Maximum Availability Architecture". His main domains of interest are: Computer Sciences, Database Architecture and Cloud Performance Management. Since 2014 he is a PhD. Candidate at the Bucharest University of Economic Studies, focusing his research on Performance management in Cloud environments.

Adriana-Nicoleta TALPEANU graduated from the Faculty of Cybernetics, Statistics and Economic Informatics of the Academy of Economic Studies in 2008. Starting 2010 she holds a MASTER degree in the field of Finance and Banking, in the area of "Managing Banking Systems". Her main domains of interest are: Banking Systems, Financial Regulations and Business Analysis. Since 2014 she is a PhD. Candidate at the Bucharest University of Economic Studies, focusing her research on the effects of the prudential rules on the banking systems.

INTENTIONAL BLANK

CONFIGURATION OF A GUIDANCE PROCESS FOR SOFTWARE PROCESS MODELING

Hamid Khemissa[1] and Mourad Oussalah[2]

[1]Computer Systems Laboratory, Computer Science Institute,
USTHB University, Bab Ezzouar Algeria
hkhemissa@hotmail.com
[2]Computer Laboratory Nantes Atlantique, Faculty of science Nantes University
Mourad.Oussalah@univ-nantes.fr

ABSTRACT

The current technology tend leads us to recognize the need for adaptive guidance process for all process of software development. The new needs generated by the mobility context for software development led these guidance processes to be adapted. This paper deals with the configuration management of guidance process or its ability to be adapted to specific development contexts. We propose a Y description for adaptive guidance process. This description focuses on three dimensions defined by the material/software platform, the adaptation form and provided guidance service. Each dimension considers several factors to develop a coherent configuration strategy and provide automatically the appropriate guidance process to a current development context.

KEYWORDS

Guidance process, Configuration process, Adaptation, Development context.

1. INTRODUCTION AND PROBLEMATIC

The software development organizations are actually confronted to difficulties regarding the development of their applications. Due to technological progress, the developer is considered nowadays as a mobile actor working in various development context using variable platforms. This trend seems interesting from a user perspective, it poses a new problem in software processes engineering. This concern denotes the adaptation ability to the possible variations of the development context. The objective is to support the process by providing software tools to model, improve, assist and automate development activities [1, 2]. For this purpose, the research in the software processes modeling have known a considerable evolution focusing on defining concepts and objectives for modeling and defining Process-Centered Software Engineering Environments [2,3,4]. They agree on the following goals like to facilitate the comprehension and communication process, to describe clearly the roles, responsibilities and interactions between users, to automate the execution of repetitive tasks that do not require the human actor intervention and to provide guidance to actors about modeling and handling a software process.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 41–57, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60604

According to the aim and orientation given to the software process, it is possible that other concepts such as strategy, organization and guidance can be described in the software process meta-model.

For this, it is necessary to assist developers and to ensure configuration management of the guidance processes [5, 6] by their ability to adapt to the current development context in respect of their usefulness. In addition, usefulness is not limited to performance criteria in the tasks accomplishment, it relates rigorously to satisfaction services offered to developers. By development context, we mean the triplet (material/software platform, developer profile, activity context). Usefulness refers to the ability of a guidance process to allow the developer to reach his objective preserving consistency and product quality in software development.

In this perspective, a rigorous guidance process targets two basic aspects: 1) The progress control of the software process development regarding the temporal constraints of the activity and the consistency of the results, and 2) the guidance interventions adapted to the specific needs within the development context in progress.

Section 2 of this paper presents a synthesis of related work and describes the current tend. Section 3 presents our approach of the Y adaptive guidance modeling, while section 4 describes the configuration management of the guidance process, as well as the Configuration of Guidance Process Meta model (CGPM) and section 5 presents the practical cases study of the adaptive guidance process. It ends with a conclusion and future prospects.

2. RELATED WORKS AND CURRENT TEND

Several process-centered environments [7, 8, 9] deal with the guidance aspect in the support of the software product development. However, the provided guidance is not often adapted to the development context profile. The orientations of the guidance are defined on the basis that the human actor, regardless of his profile (qualifications and behaviour), has a central role in the progress of the development process.

Among this new generation of the software process engineering, we can invoke the following meta-models and modeling environments: SPEM [10] and APEL [8] considered as the most representative in the software process modeling, RHODES [7, 11] that uses basic concepts closest to those introduced by the proposed approach.

SPEM meta-model introduced the concept of "Guidance". According to SPEM, the guidance is a describable element, which provides additional information to define the describable elements of modeling. However, the proposed guidance is not suitable to the development context's profile (role, qualifications and behaviour). The guidance is rather defined in an intuitive way. ADELE/APEL proposes a global guidance of proscriptive type without considering the development context profile and automates part of the development process using triggers. RHODES/PBOOL+ uses an explicit description of a development process. The activities are associated to a guidance system with various scenarios of possible realization.

An effective support to software process depends on several factors, in particular the personalization and adaptation factor. The definition of a process with an active guidance for automation and coherency control would be effective if it can be adapted to each development

context. The platform, tasks and developers factors may considerably vary. An improved productivity and development process adaptation would be possible, if a process can be adapted considering the fact that these factors can be exploited.

Actually, there are Process Centered Software Engineering Environments (PSEE) allowing changes during the execution, where the developer is in a position to predict the execution model before running it. However, these models do not provide appropriate performance models. Some PSEEs use a guidance description structured in phases like prescribing systems or proactive systems to control the operations carried out by the developer. Nevertheless, they are essentially limited to the adaptive guidance aspect to current development context.

Taking into account specific criteria for an adaptive guidance, we have classified these limits through several criteria describing explicitly the basic concepts linked to the adaptive guidance [5, 6, 12]. To realize the effectiveness of configuration concept of the guidance process supported by its adaptation ability to current development context, we refer to the studied meta-models and modeling environments [12, 13, 14, 15, 16]. The selected criteria are defined by:

- Global guidance core: The basic guidance is defined as a global orientations core regardless the profile of both the activity and the actor.
- Developer profile oriented guidance: the guidance orientations are defined on the basis that the human actor, regardless his profile, has a central role in the progress of the development process.
- Context development guidance: The selection of the appropriate type of guidance is more often not adapted nor suitable to a current context.
- Guidance types: the selection of guidance types remains defined in a manual and in an intuitive way. It depends on the experience and on the informal personality of the project manager.
- Configuration of guidance process: the guidance functions are defined and offered on the basis that the human actor always operates in a uniform development context.

To respond to these limits, one currently tries to offer more flexibility in the language of software process modeling. This tendency results in the idea to define interventions of direct and adaptive guidance in particular contexts during the progress of software process. In considering the principal limitations of PSEEs and essential characteristics of our approach in particularly the context adaptation aspect, a comparative table of the studied meta-models is as follows.

Table 1. Comparative table of the studied meta-models.

Meta model / Criteria	ADELE/APEL	RHODES / PBOOL+	SPEM
Global guidance core	Global	Global	Global
Developer profile oriented guidance	Not adapted	Considered strategy Model	Not adapted
Context development guidance	Not adapted	Adapted	Not adapted
Guidance types	Not invoked	Associated with a specific guide system	Intuitive selection
Configuration of guidance process	Not covered (Single Platform)	Not covered (Single Platform)	Not covered (Single Platform)

The current tendency is that developers would like to have integrated environments that are suitable to specific needs according to the development context factors. However, despite the necessity imposed by technological evolution, the provided efforts to develop such environments remain an insufficient contribution. This generation of guidance environment still interests researchers in defining new concepts and objectives of the software process modeling [4, 17, 18].

Our work proposes an approach to define adaptive guidance modeling in software process. The proposed approach concepts are described through a meta-model denoted CGPM (*Configuration of Guidance Process Meta model*). The information provided must be adapted to the development context profile. They must guide the developer during the software process development through suitable actions and decisions to undertake with corrective, constructive or automatic intervention [12]. Three dimensions defined by the development context, the adaptation form and the provided service explicitly describe our guidance process adaptation.

3. THE ADAPTIVE GUIDANCE IN Y

A guidance process may be processed in many different ways according to the perspective guidance to provide interveners with development context. Thus, there are generally several possible guidance models, each of them with a particular relevance and need. This vision denotes the configuration of guidance process, and its ability to adapt to its development context. The configuration concept describes its capacity to adapt to the intrinsic variations of required conditions in terms of usefulness [16, 19, 20].

In this context, we propose a description in Y of the adaptive guidance. This description will focus on the three considered dimensions. Each dimension considers several factors to deduce automatically the appropriate guidance service according to the current context. Schematically, we describe it as follows:

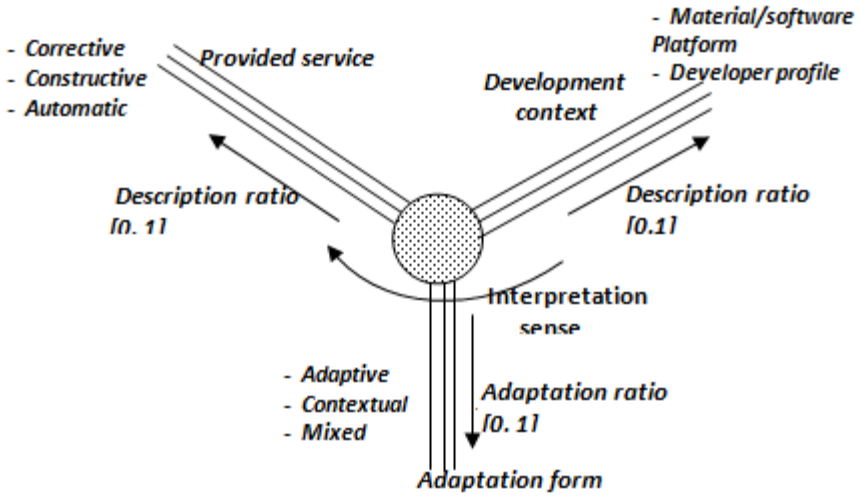


Figure 1. Adaptive guidance description in Y.

The principle of our approach is to generate, from the development context related to the specific data for each defined models according to the retained adaptation form, the guidance interventions (corrective, constructive, automatic) adapted to the current development context.

3.1. The basic conceptual model description

The conceptual model highlights the guidance process aspect through the adaptation form described by the inherent relationship between the three considered dimensions.

1. The development context

A guidance intervention is provided according to an object or set of objects. An object is associated to the following:

- The material/software platform: described by the computing resources, the software services as well as the interaction and communication modes.
- The activity context: models the structure and the workflow, they are defined by a progression mode in the activity ensuring that all tasks can be performed under control in a preset order established by the designer and a temporal progression mode specifying deadlines for completion.
- The developer profile: defines the specific properties of each developer. These properties can be either static or dynamic. The static aspect refers to the user characteristic as his role, his business competence and his familiarity with the software process. The dynamic aspect refers to the behaviour of using the guidance service, by the fact to execute, to define or to complete the software process resource and the user's reaction to a guidance message.

The description performance rate of these factors is evaluated by considering each identified object as concepts, principles, procedures, and resources. These guidance objects represent the basis of different guidance interventions related to a particular situation. This performance serves as the selection of the adaptation form to retain and guidance service to provide to the user.

2. The adaptation form

Each guidance intervention is done according to the retained adaptation form. It relates to a specific situation described by the development context description. Our modeling approach allows the following guidance adaptation forms:

- **Contextual guidance**: intervention is provided dynamically according to the material/software platform and activity models and the state of the process. The adaptation rate is related to the model description rate of the activity and the material/software platform. The guidance intervention does not consider the developer model (e.g. to avoid inconsistency during the affectation of a resource).
- **Adaptive guidance**: intervention is provided according to the developer model and the material/software platform specificity (e.g.: the user asks for explanations on his choice). The adaptation rate is related to the developer and material platform models description.
- **Mixed guidance**: intervention is provided according to the development context (e.g.: to guide the developer on the sequencing principle during the software process progression). This form describes the highest adaptation rate. This rate is evaluated on the basis of the developer, activity and material/software platform models description.

The adaptation form performance is described by a strong coupling between the development environment and guidance process. It determines the relevance and precision of the guidance provided to developers.

This criterion is directly related with the adaptive guidance process concept. Through a strong coupling, the process would deduce the guidance service and can therefore extract useful and helpful information to the user.

3. The provided service

The guidance process offers several service types in relation to a defined context by the current development and adaptive form. The provided guidance services are corrective, constructive or automatic order.

- **Control and taking corrective initiative:** protect the user of his own initiatives when they are inadequate under progress.
- **Control and taking constructive initiative:** the ability to take positive initiatives, executing and combining the performance of operations without the user intervention.

The guidance adaptation performance associated to a development environment is done by enrichment or reduction of the possible offers of the guidance. Among these offers, we have:

- **The directive guidance:** to show the developer how to execute a task by an adaptive control of the guidance system, specifying the steps of an activity or the whole process development.
- **Retroaction**, to offer the developer more information on the activities context (e.g. new available resources) or on the progress state of his work (progression of an activity).
- **Explanation**, to offer explanations about a guidance object at the request developer. (e.g. the activities coordination of the software process).
- **Reminding**, to remind the developer some principles or procedures on the sequencing of the activities or their activation conditions when the system detects a conflict or inconsistency.

- **Automatic guidance:** analyze the impact projection to define the solution to consider in order to avoid deadlocks or delays, by the fact to start, suspend, discontinue or continue ongoing actions to avoid conflict.

These guidance services can be combined. They may be temporary, permanent or left under the developer control.

The usefulness rate is evaluated by the degree of the performance description of the development context and adaptation form.

4. THE CONFIGURATION MANAGEMENT OF A GUIDANCE PROCESS

The management discipline within the software engineering process allows to develop a reference basis to any planned configuration. The configuration management is a complex activity that covers all the steps in a software process, from the conceptual aspects to its configuration and installation. It includes the initiation techniques and changes control that manage the software process progress [14, 15, 16].

The role of configuration management is to identify the configuration items, manage the sequence and ensure the evolution and any likely change. These activities are usually treated in different ways depending on the nature and specificity of the objective [15, 16, 21, 22].

In this context, the configuration management of a software process and particularly a guidance process is a configuration mechanism rigorously linked to the evolution of all development context factors. This evolution process is influenced by the temporal and contextual elements. Therefore, it is necessary to define the participants' elements and triggers the configuration of a guidance process, and clarify their functional interdependencies.

We describe graphically the configuration basis of our guidance process defined through five interrelated elements in figure 2:

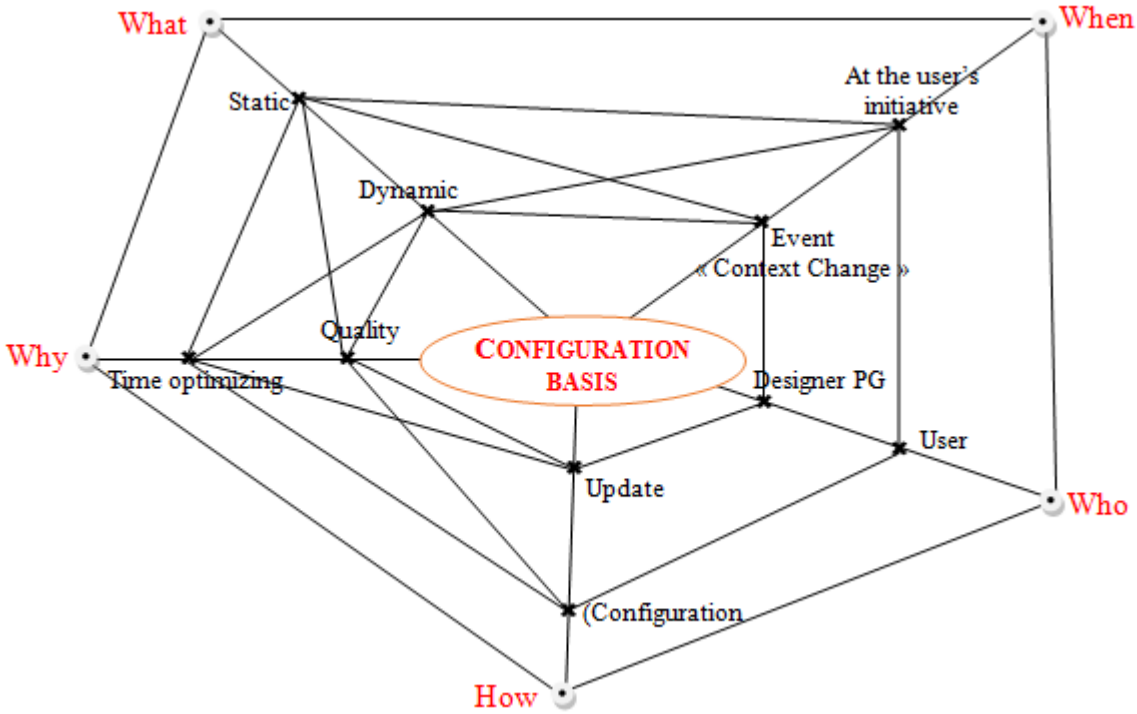


Figure 2. The configuration basis of guidance process.

This evolution basis allows, each time, to answer to any manifest changes provoked statically or dynamically in order to design the most appropriate guidance process related to the current situation. The configuration management of guidance process is useful both for time optimizing and for the guidance process quality.

The (Re) configuration is invoked by the basic guidance process actors, namely the guidance process designer or at the user's request, with different interests ensuring an equilibrium between the user's needs in terms of guidance, with either static adaptation at the guidance conceptual level or dynamic process related to the development context change. The following illustration defines the basis description of the guidance invocation.

The configuration process deployment involved represents the core of the configuration management. It describes the steps and the contextual sequencing that should be undertaken to ensure this goal. Therefore, the deployment cycle of a guidance process configuration is explained via all activities related to the configuration management and their functional interdependencies (see figure 4.).

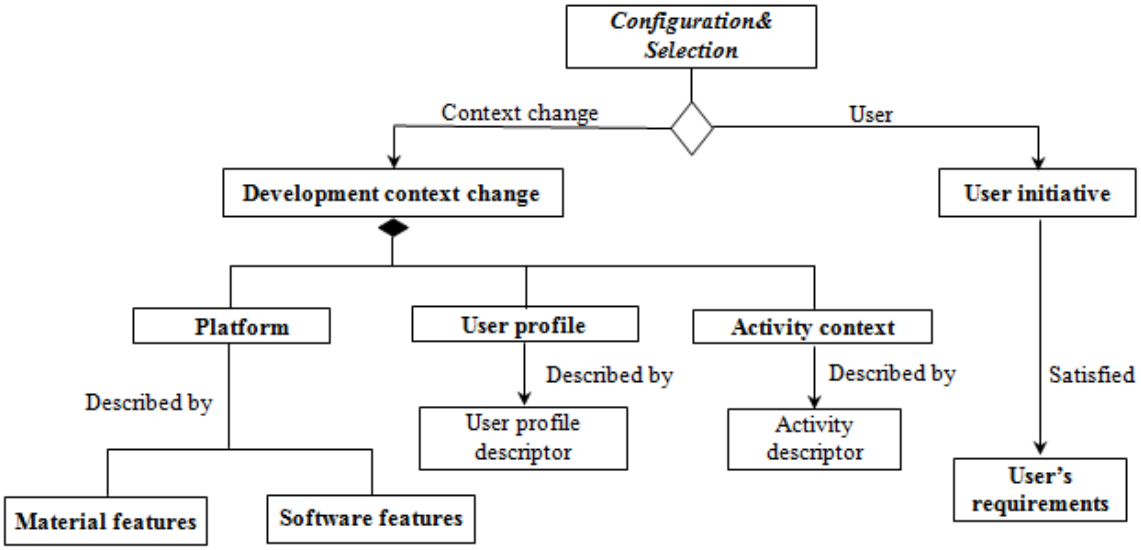


Figure 3. The basis description of the guidance invocation.

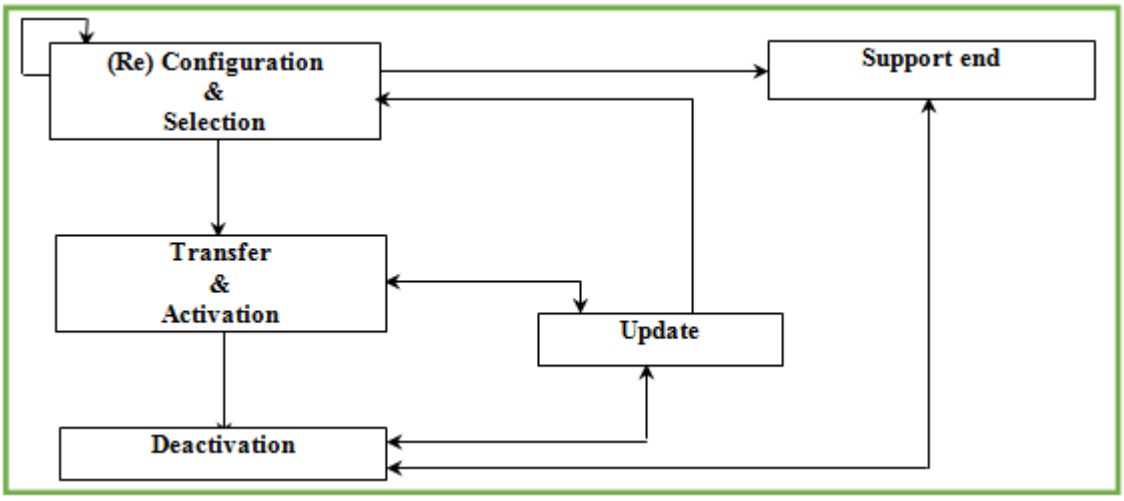


Figure 4. The guidance process configuration.

- The first activity concerns the configuration and selection of the most coherent composition, which best responds to the defined development context. Its objective is to offer the most appropriate guidance to support the current situation.
  For this, we should first deal with the dependencies between the guidance process and description of the development context, to specify the expected guidance service and constraints to be respected.

- The transfer and activation activity consists in transferring the guidance support. It is invoked at the initiative of the guidance process designer or at the request of the user. Once installation and transfer problems solved, the activation operation role is to put the guidance support in the active state, ready to be executed.

- The update activity is to resume, adjust or to complete the guidance support configuration. It is invoked at the user's initiative or by the guidance process designer following a change of the development context.

- This activity is almost similar to the (re) configuration, with the difference that the guidance support is already installed. Finally, the implementation of the update uses the deactivation activity.

- The "support end" activity is relative to the end of guidance service offered to users of the software process in progress.

- Deactivation activity is essential before any configuration, update or support end activities. Its role is to deactivate the support guidance component so that we can perform the likely changes of some support components.

The configuration management is an application developed based on a set of constraints and / or preferences described by the development context factors. If one of these constraints and / or preferences causes a notified event of change then it may be that we do evolve the current configuration either statically (inactive process) or dynamically (without interrupting the guidance process progress).

4.1 The Configuration of Guidance Process Meta Model

Our modeling approach CGPM (***Configuration of Guidance Process Meta model***) is defined with reference to studied PSEEs features. The aim of our approach is to better respond to the factors of the development context defined by the three dimensions, the development context, adaptation form and offered guidance. Each dimension is described through several basic factors to develop a coherent configuration strategy and provide the most appropriate guidance support to the current development context.

In this context, our meta-model is based on a conceptual model of a software process enriched by a configuration model adapted to the guidance process. The configuration model detects any event of the development context evolution and launches the configuration of guidance process.
The role of the first activity "configuration and selection" is to prepare the guidance support to the current situation. The second service "transfer and activation" is to put the guidance support in the active state. The third service "Update" is to provide a way to modify and / or complete the guidance configuration following a development context change.

The configuration strategy evolves according to the political autonomy given to the guidance system respecting the application conditions. The implementation of this policy is based on an adaptation mode expressed by a set of rules of ECA form (Event, Conditions, Actions). For each execution in the development context, if required conditions related to the context and the adaptation form then launch guidance configuration strategy to generate the most appropriate guidance service.

The proposed meta-model aims at generating the adapted guidance interventions to the development context in relation to the considered factors and specific data for each defined dimension (see Figure 6).

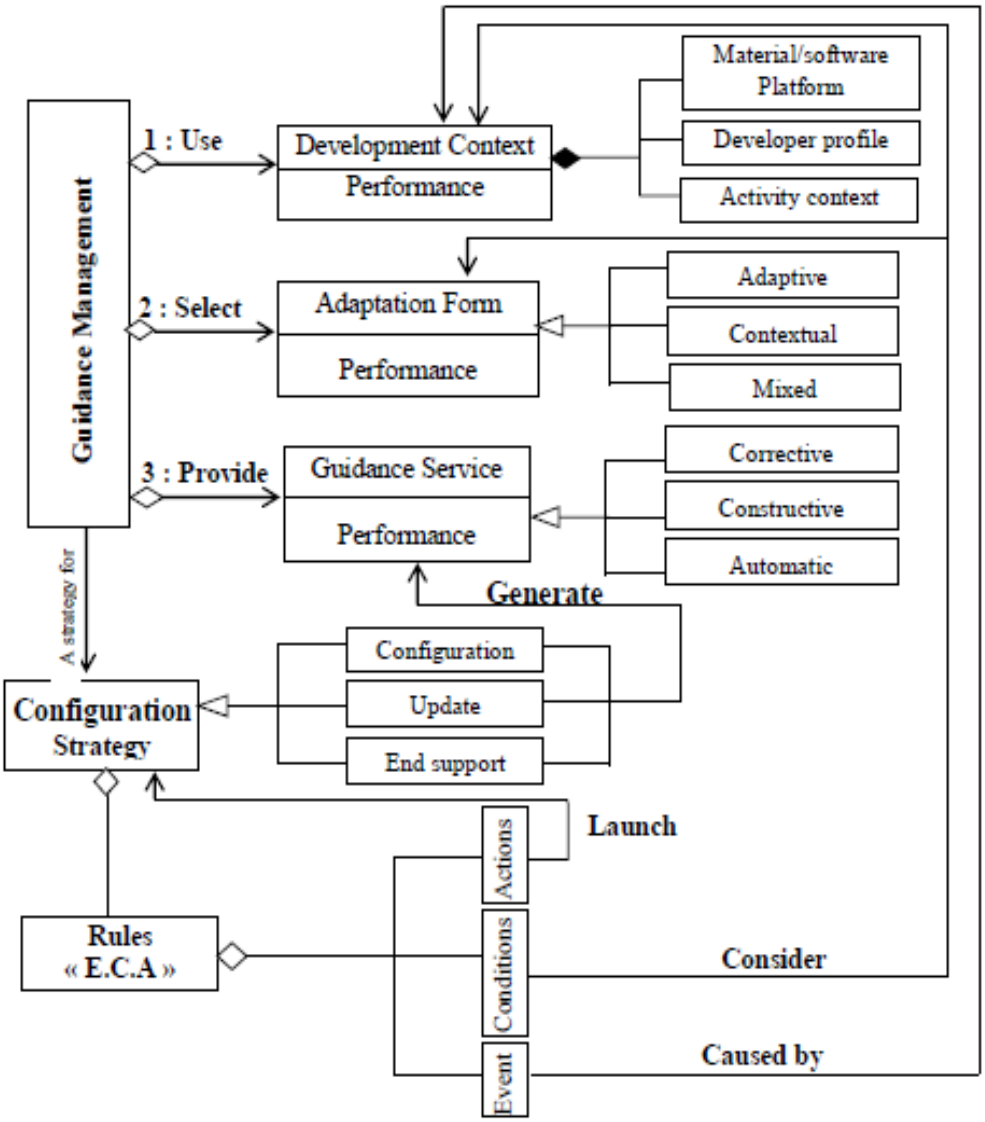


Figure 6. Configuration of Guidance Process Meta model.

5. THE PRACTICAL INTERPRETATION

Considering the software process model "Activity test", the process "Activity test" in the software development is composed of several types of tests such as; Integration test and Unitary test. Each receives as input a test plan and provides a test report. For each type of test, there is a manager, responsible of the execution.

A performing tree given in Figure 7 describes the activity process «Activity test». We notice that the activity test starts the execution of sub activities "Unitary test" then "Integration test". The unitary test launches in parallel the execution of tasks "Test unit".

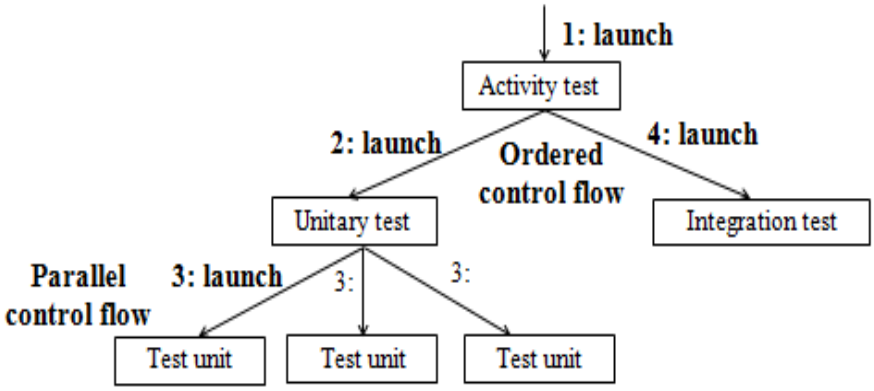


Figure 7. The activity test process

To simplify our example, we consider the execution process of the unitary component test. The application of the activity “Unitary test", requires the list of components. It calls the tool that will create the necessary environment to carry out the actual execution of the “Unitary test", as the state diagram, the test variables, etc. ... the activity "Unitary test" launches in parallel the different tasks "Test unit" where an event signals the beginning of the “Test unit” execution. Finally, the ended event is broadcast.

The adaptive execution process of the activity "Unitary test", regarding our adaptive guidance approach is described in Figure 8.

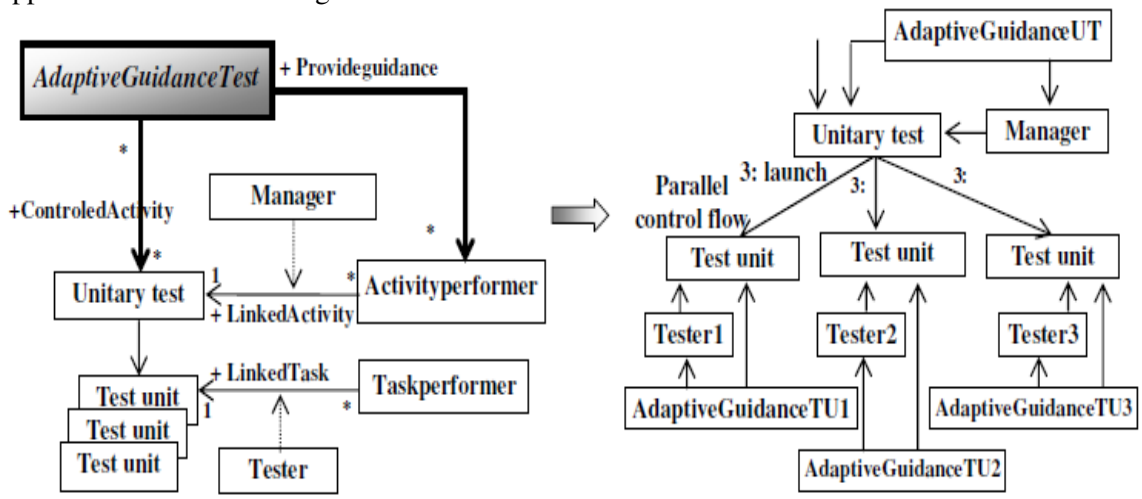


Figure 8. The adaptive execution process “ Unitary test”.

The adaptive guidance is linked to the manager or to each tester according to the current development context profile defined by its material/software platform, the activity context and developer profile. We explain this adaptive approach through the following situation; the testers have the same role “test unit” with identical activity model. However, the developer’s qualification and the material/software platform specificity differ from one development context to another. According to the current context, it happens to enrich or reduce the appropriate guidance intervention or generate several possible forms of guidance services.

We consider three situations with tester's qualification defined respectively as high, medium, and low. The study case is related to launch the test unit without having all the input data, by selecting the appropriate test variables and generating the unit test report. The adaptive guidance process related to each qualification case is described as follows:

1. For a development context with high qualification tester and a high material platform performance: the tester starts the test unit process on the basis of the defined plan by taking his proper initiatives. The development context evaluation allows deducing the adaptation form to retain and the guidance service to provide. In this case, we adopt the adaptation guidance form and the provided guidance intervention is thus of a corrective order. The corrective intervention is provided to inform the manager of the setback and remind him of the corresponding unitary test diagram. The manager remains free to take into account the intervention.

2. For a development context with an average skill tester and an acceptable material/software platform performance: the tester starts the test unit process by applying rigorously the defined test plan. The evaluation of such context results in a contextual guidance form and the provided guidance intervention is thus of a constructive order. The guidance system analyzes the current context of the task, evaluates the impact and consequence of the delay caused in comparison with possible margins and offers a possible solution to the manager (solution: the guidance proposes to cancel the launch of the current test unit and generates a new execution plan according to the rate of delay and possible margins). The construction solution is not definite; the manager should validate it.

3. For a development context with a low qualification tester and a reduced material/software platform performance: the tester starts the test unit process by applying reliably the defined test plan. The development context evaluation results in a mixed guidance adaption form and the provided guidance intervention is thus of an automatic order. The guidance system analyzes the current context, cancels the launch of the "test unit" task, evaluates the impact and consequence of the delay caused in comparison with the possible margins and automatically updates the execution plan of "unitary test" activity.

5.1. The digital application

The practical definition of the adaptive guidance type for each considered profile is deduced through a quantitative process of the factors in relation to the basic models (materiel/software platform, activity context, developer profile). The considered example is processed as follows.

Each profile is semantically described in table (see Table 2). The project manager under the specification of an ongoing project [17] determines the semantics evaluation and the weighting. To scan the semantics evaluation, we associate the weighting related to the interest granted to each attribute.

Table 2. The profiles evaluation.

Development context	Factors	Context Profile 1	Context Profile 2	Context Profile 3	Context Profile 4	Context Profile 5	W[i]
Material/ software Platform	Development System Constraint	Low	Medium	High	High	Low	**P2**
	Software Tools	Low	Medium	Low	Provided	Provided	**P1**
	Memory Constraint	High	Medium	Medium	High	Medium	**P3**
Developer Profile	Role	No effect	Classic	Critique	Critique	No effect	**P4**
	Competence	High	Medium	Low	Low	High	**P1**
	Familiarity with Software Process	Quite Acceptable	Medium	Low	Low	Acceptable	**P1**
	Behavior for guidance	Most Appropriate	Satisfying	Inadequate	Adequate	Inadequate	**P2**
Activity Context	Density of tasks in the activity	Acceptable	Acceptable	Acceptable	Acceptable	Acceptable	**P3**
	Complexity Level	Medium	Medium	Medium	Medium	Medium	**P2**

With $W[i] \in [1, 5]$. Where Pi represents the computing value.

Considering the similar principle such as the COCOMO model [22], the quantification of each profile's feature is on the data range] 0, 2 [, (see Table 3). This quantification is usually based on the impact of each feature.

It is usually done through three levels, described by high, medium or low contribution, applying the following rules:

1: middle order impact / <1: positive impact / >1: negative impact.

In this stage of profiles' process, and in case of simple profiles' samples, we can proceed to associate each considered development context profile to the appropriate guidance adaptation form and guidance service.

The guidance profile (GP) associated to each profile class is based on the following formula:

$$GP(Px) = \Sigma Ai\ Wi / 2*\Sigma Wi \text{ avec } i=1 \text{ to } n$$

With :

Ai: the feature value.

Wi: the associated weighting.

Px : the associated profile.

The adaptation form and the guidance profile of each considered development context profile based on the evaluation of each model and GP value is given by (see Table 4).

Table 3. The profiles quantification

Development context	Factors	Context Profile 1	Context Profile 2	Context Profile 3	Context Profile 4	Context Profile 5	W[i]
Material/ software Platform	Development System Constraint	0.75	1.00	1.30	1.30	0.75	1
	Software Tools	1.20	1.00	1.25	0.80	0.80	3
	Memory Constraint	1.40	1.00	1.00	1.60	1.00	2
Developer Profile	Role	0.40	1.00	1.90	1.70	0.40	4
	Competence	0.20	1.00	1.70	1.70	0.25	3
	Familiarity with Software Process	0.40	1.00	1.60	1.60	0.30	3
	Behavior for guidance	0.20	0.80	1.70	0.75	1.60	1
Activity Context	Density of tasks in the activity	0.80	0.80	0.80	0.80	0.80	2
	Complexity Level	1.00	1.00	1.00	1.00	1.00	1

Table 4. The associate guidance profile.

	Context Profile 1	Context Profile 2	Context Profile 3	Context Profile 4	Context Profile5
Associated Adaptation form	Adaptive	Contextual	Mixed	Mixed	Adaptive
Guidance Profile (GP)	0.333	0.485	0.721	0.673	0.315
Associated guidance profile	Corrective	Constructive	Automatic	Automatic	Corrective

It should be noted that the value of GP ranged from 0 to 1 and the range associated with each type of guidance is defined by the fixed limits to each guidance type. If the range of corrective guidance is fixed between 0 and 0.35 and the range of the constructive guidance is between 0.36 and 0.65, we automatically associate a corrective guidance to profile P1 and P5, and a constructive guidance to profile P2 and automatic guidance to profile P3and P4.

However, in case of a very important population, and for the aim of optimizing profile classes, it is recommended to proceed in the gathering and classification of the provided development profile and reasoning in relation to generated classes.

6. CONCLUSION

Our main purpose in this article is to propose a configuration management of a guidance process for software process modeling. This configuration is highlighted through a Y description of our adaptive guidance. This description will focus on three dimensions defined by the material/software platform, the adaptation form and the provided service. Each dimension considers several factors to deduce automatically the appropriate guidance process according to the current development context. The proposed approach concepts are described through a meta-model denoted CGPM (*Configuration of Guidance Process Meta model*). The proposed meta-model aims to generate the adapted guidance support to the development context in relation to the considered properties and specific data for each defined model.

The contribution of this approach is to institute the configuration management concepts, in order to be able to proceed, every time, with the adaptation of the guidance support to the various changes in the development context to ensure its adaptive deployment to software process.

The configuration strategy in the adaptation is done by the dynamic deployment of the provided guidance process. Intuitively, we consider the defined guidance process configuration through the sequencing of contextual activities. The guidance strategy evolves according to the political autonomy given to the guidance process respecting the application conditions. The implementation of this policy is based on an adaptation mode expressed by a set of rules of ECA form.

A perspective to this work concerns, at first, the necessity to estimate the productivity and cost due to the adaptation of guidance system.

In a second step, we will ensure the development of semantic rules that allow swapping through different guidance profiles, either statically by adjusting the guidance parameters or dynamically through the development context changes.

REFERENCES

[1] Ivan Garcia and Carla Pacheco « Toward Automated Support for Software Process Improvement Initiatives in Small and Medium Size Enterprises ». Book chapter. Software Engineering Research, Management and Applications 2009 Volume 253/2009, pp. 51–58. c_ Springer-Verlag Berlin Heidelberg 2009. ISBN: 978-3-642-05440-2.

[2] Kirk, D.C, MacDonell, S.G., & Tempero, E. 2009 Modeling software processes - a focus on objectives, in Proceedings of the Onward, 2009. Conference. Orlando FL, USA, ACM Press, pp.941-948.

[3] Benoît COMBEMALE, Xavier CRÉGUT, Alain CAPLAIN et Bernard COULETTE. Towards a rigorous process modeling with spem. Dans ICEIS (3), pages 530–533, 2006

[4] Hamid Khemissa, Mohamed Ahmed-Nacer, Mourad Daoudi, 2008. A Generic assistance system of software process. In International Conference on Software Engineering: Software Engineering. SE 2008, Feb 12-14-2008, Innsbruck, Austria.

[5] Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Souchon, N., Bouillon, L., Florins, M., Vanderdonckt, J.: Plasticity of User Interfaces: A Revised Reference Framework. In: TAMODIA 2002 (2002).

[6] Joëlle Coutaz, EICS '10. User interface plasticity: model driven engineering to the limit!. Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems. June 2010.

[7] Coulette B., Crégut X., Dong T. B. T. and Tran D. T., "RHODES, a Process Component Centered Software Engineering Environment", ICEIS2000, 2nd International Conference on Enterprise Information Systems, Stafford, pp 253-260, July 2000.

[8] Jacky Estublier, Jorge Villalobos, Tuyet Le Anh, Sonia Jamal-Sanlaville and German Vega. An Approach and Framework for Extensible Process Support System. In Proceedings 9th European Workshop on Software Process Technology (EWSPT 2003), Helsinki, Finland, 2003-09-01.

[9] Hans-Ulrich Kobialka, « Supporting the Software Process in A Process-centered Software Engineering environment », Upgrade-cepis.org/issues/2004/5/upgrade-v VOL; V n° 5 October 2004.

[10] OMG. Inc. Software and System Process Engineering Meta-Model Specification version 2.0: Formal/2008-04-01.

[11] Tran Hanh Nhi, Bernard Coulette, Xavier Crégut, Thuy Dong Thi Bich, Thu Tran Dan. Modélisation du méta-procédé RHODES avec SPEM. Dans : Recherche Informatique Vietnam-Francophone (RIVF'03), Hanoi, Vietnam, 2003.

[12] Hamid khemissa, Mohamed ahmed nacer & Mourad Oussalah «Adaptive Guidance System for SPEM ». The First International Conférence on Information Technology Convergence and Services; ITCS, SIP, JSE 2012 pp. 429-441, Bangalore, India.

[13] Hamid Khemissa, Mohamed Ahmed-Nacer, Mourad Oussalah «Adaptive Guidance based on Context Profile for Software Process Modeling». Information Technology and Computer Science, , July 2012 in MECS 2012.

[14] C. Lueninghoener. "Getting Started with Configuration Management. ;login: issue: April 2011, Volume 36, Number 2" (PDF). Retrieved 2012-11-23.

[15] IEEE Std-828-2012 IEEE Standard for Configuration Management in Systems and Software Engineering. 2012. doi:10.1109/IEEESTD.2012.6170935. ISBN 978-0-7381-7232-3.

[16] Ali, U., Kidd, C. Barriers to effective configuration management. Application in a project context: An empirical investigation, International Journal of Project Management, 2013b; http://dx.doi.org/10.1016/j.ijproman.2013.06.005.

[17] Grambow, Gregor and Oberhauser, Roy and Reichert, Manfred (2011) Enabling Automatic Process-aware Collaboration Support in Software Engineering Projects. In: Selected Papers of the ICSOFT'11 Conference. Communications in Computer and Information Science(CCIS).

[18] Clarke, Paul and O'Connor, Rory (2011) An approach to evaluating software process adaptation. In: 11th International SPICE Conference on Process Improvement and Capability dEtermination, 30 May - 1 jun 2011, Dublin, Ireland. ISBN 978-3-642-21233-8.

[19] Sottet, J.-S., Calvary, G., Coutaz, J., Favre, J.-M. A Model-Driven Engineering Approach for the Usability of User Interfaces. In Proc. Engineering Interactive Systems (EIS2007), J. Gulliksen et al. (eds), LNCS 4940, (2007), 140-157

[20] Ferry, N. Hourdin, G., Lavirotte, S., Rey, G., Tigli, J.- Y., Riveill, M. Models at Runtime: Service for Device Composition and Adaptation. In 4th International Workshop Models@run.time, Models 2009(MRT09).

[21] Lindkvist, C., A. Stasis, and J. Whyte, Configuration Management in Complex Engineering Projects. Procedia CIRP, 2013, Volume 11(0): p. 173-176.

[22] Barry W. Boehm, Chris Abts, A. Winsor Brown, Sunita Chulani, Bradford K. Clark, Ellis Horowitz, Ray Madachy, Donald J. Reifer, Bert Steece, 2009. Software Cost Estimation with COCOMO II. Prentice Hall Edition, ISBN: 0137025769, 978013702576.

AUTHORS

Hamid Khemissa is a full associate professor at Computer Systems Department, Faculty of Electronics and Computer Science, USTHB University, Algiers. He is member of the software engineering team at computer system laboratory LSI, USTHB. His current research interests include Software Process Modeling and Software Modeling Guidace.

Mourad Chabane Oussalah is a full Professor of Computer Science at the University of Nantes and the chief of the software architecture modeling Team. His research concerns software architecture, object architecture and their evolution. He worked on several European projects (Esprit, Ist, ...). He is (and was) the leader of national project (France Telecom, Bouygues telecom, Aker-Yard-STX, ...). He earned a BS degree in Mathematics in 1983, and Habilitation thesis from the University of Montpellier in 1992.

INTENTIONAL BLANK

USING MUTATION IN FAULT LOCALIZATION

Chenglong Sun and Tian Huang

Institute of Software, Chinese Academy of Sciences, Beijing, China
suncl@ios.ac.cn huangt@ios.ac.cn

ABSTRACT

Fault localization is time-consuming and difficult, which makes it the bottleneck of the debugging progress. To help facilitate this task, there exist many fault localization techniques that help narrow down the region of the suspicious code in a program. Better accuracy in fault localization is achieved from heavy computation cost. Fault localization techniques that can effectively locate faults also manifest slow response rate. In this paper, we promote the use of pre-computing to distribute the time-intensive computations to the idle period of coding phase, in order to speed up such techniques and achieve both low-cost and high accuracy. We raise the research problems of finding suitable techniques that can be pre-computed and adapt it to the pre-computing paradigm in a continuous integration environment. Further, we use an existing fault localization technique to demonstrate our research exploration, and shows visions and challenges of the related methodologies.

KEYWORDS

Software testing, fault localization, continuous integration, mutation testing

1. INTRODUCTION

Software serves every corner of our lives. Nevertheless, software failures nowadays are still common and it is extremely difficult to thoroughly avoid software failures, even in final releases of software. Most software failures are caused by mistakes made by programmer in coding and debugging is an important activity in software engineering with the intent of locating and removing faults from programs. Conventionally, a debugging process involves three tasks, fault localization, fault repair and retesting of the revised program. Among them, fault localization is the most difficult and time-consuming task, and is often a bottleneck in the debugging process.

To facilitate the fault localization task, automatic fault localization techniques have been invented. Generally speaking, these techniques automate the identification of the suspicious code that may contain program faults. Tarantula [1] calculates the ratio of passed runs and ratio of failed runs that executes a program statement, and estimates the suspiciousness of that program statement correlated with program failures. A ranked list of program statements is constructed in descending order of the estimated suspiciousness of each statement. These kind of techniques, which contrasts the program spectra of passed and failed runs to predicate the fault relevance of individual program entities, is called coverage-based fault localization (CBFL).

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 59–66, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60605

CBFL has received much attention due to its applicability and is a popular research area on the exploration of effective evaluation formulas to assess the suspiciousness of program entities. The techniques in CBFL are of low-cost at ranking program entities because only testing results and program spectrum for software testing are taken into account [2][8]. A typical technique in CBFL first selects a set of program entities, and then collects the execution statistics of them for both passed and failed runs. By contrasting the similarities between two such sets of statistics for each entity, it estimates the extents of the program entities correlating to faults, and ranks the program entities accordingly.

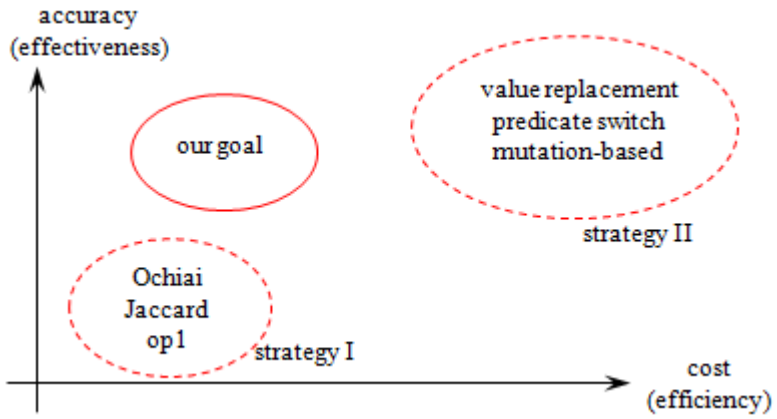


Figure 1. The cost and accuracy of different fault localization techniques

Apart from the lightweight CBFL techniques, there are many approaches aiming for high accuracy at the expense of high cost. Value replacement [10] searches for interesting value mapping pair, in which a corresponding replacement at a program site during the execution of a failing run results in correct output. Using such value mapping pairs, program statements are ranked in their likelihood of being faulty. Mutation-based fault localization techniques [7] mutate statements in programs systematically and estimate their likelihood of being faulty based on both coverage and how mutations affect the outcome of test runs. Comparing with CBFL techniques, such techniques make use of additional information like variable values, and experiments show significant improvements in fault localization effectiveness over the former.

Figure 1 illustrates the cost and accuracy differences of the above two strategies. Techniques following strategy I (CBFL techniques like Jaccard [1]) have low computation cost and relatively low locating accuracy. Techniques following strategy II have relatively high computation cost and high locating accuracy. Aiming for both low-cost and remarkable accuracy, in this paper, we integrate properties of the above two strategies to propose a low-cost and high-accuracy fault-localization direction.

Generally speaking, remarkable accuracy comes from sufficient computation to capture enough useful information to facilitate fault localization. As a result, we cannot reduce necessary computation but instead use pre-computing to move the time-intensive computing task at the testing moment to the code phase [11]. Thus, we can rapidly return fault localization results. This is the research problem proposed in this paper. We propose a two-step strategy to solve this problem: 1) To find fault localization techniques, which accuracy can be reserved by pre-computing; 2) To adapt fault localization techniques to the continuous integration scenario to get rapid response.

The contributions of this paper have at least two aspects. First, we propose a pre-computing paradigm in a continuous integration environment in order to effectively and efficiently run existing fault localization techniques. Second, we illustrate our preliminary idea by adapting an existing high-accuracy fault localization technique in the proposed paradigm.

The rest of this paper is organized as follows. Section II reviews related work. Section III motivates our work and Section IV explores this research direction, and shows visions and challenges of the research. Section V concludes this paper.

2. RELATED WORK

Coverage-based fault-localization (CBFL) techniques are one category of the most representative practices of fault localization. Jones et al. proposed Tarantula, which uses the propositions of passed and failed executions to calculate the suspiciousness of a statement to be faulty. Other fault localization techniques include Jaccard [1], Ochiai2 [3] and so on. These techniques are similar to Tarantula except that they use different formulas to compute the suspiciousness of statements. Apart from these statement-level techniques, there are many predicate-based techniques. Liblit et al. [4] developed CBI, which uses the number of times a predicate being evaluated true in passed and failed runs along with the specificity and sensitivity of the predicates in order to estimate the suspiciousness of predicates. SOBER [5] compares the distributions of evaluation biases between passed runs and failed runs to compute the suspiciousness of predicates.

Theoretically, Naish [6] showed that the optimal effectiveness of CBFL techniques is relatively not higher than that of Op1. However, the effectiveness of Op1 may not always be satisfactory. For example, in some programs, examining 200 lines of code to locate a fault can be too much for a programmer [6].

In order to achieve high accuracy in fault localization, additional information is referenced. X. Zhang et al. [9] proposed to forcibly switch a predicate's outcome at runtime and alter the control flow to estimate the position of fault statement, to accurately locate fault. Jeffrey et al. [3] proposed a value replacement technique, by looking at data-flow information and so on, which computational complexity is even higher. Papadakis and Le-Traon [7] proposed a mutation-based fault diagnosis approach, which uses mutation operators to generate different program versions, monitor testing outcome of them, and use the program version having most similar performance with the version in hand to estimate a fault's location.

Generally speaking, high accuracy acquired in the above approaches means high computation costs. For example, the value replacement technique [7] is quite slow and its IVMP searching process may last for several hours. Such slow responses are unacceptable in practice. In this paper, we plan to use pre-computing to distribute heavy computation to the idle phase, to create high accuracy and low-cost fault localization approaches.

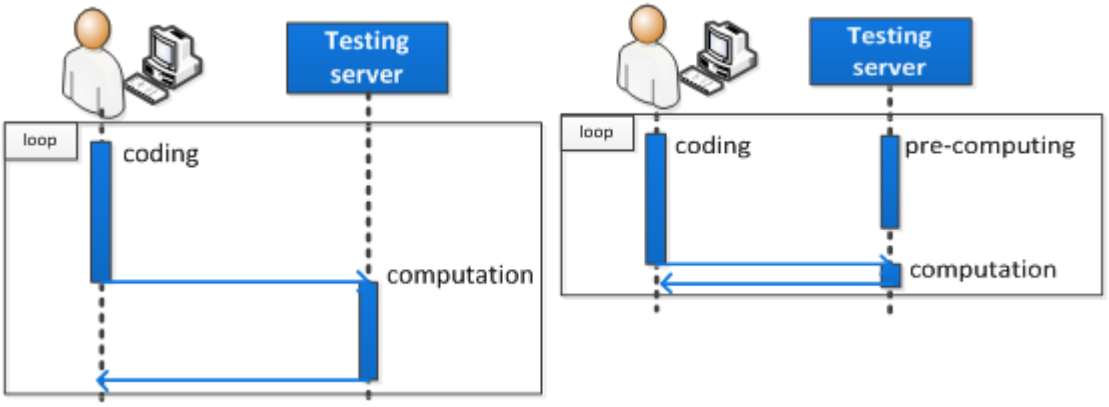
3. MOTIVATION

We revisit different fault localization techniques and quantify their cost and effectiveness in Table I. We collect data both from the reported result from the original paper of the mentioned techniques and from our experiment. We use the running time of a technique to output a fault

localization result to measure its computation cost, and use the percentage of faults located in a date set within 5% of code examination efforts in each program to measure the accuracy of the fault localization technique.

Table 1. The cost and accuracy of different fault localization techniques (quantified)

	Avg. running time (seconds)	Avg. accuracy (% of faults located when examining 5% of code)
Group I / Strategy I (e.g., CBFL) (low-cost, low accuracy)	< 5	< 30%
Group II / Strategy II (e.g., value replacement) (high-cost, high accuracy)	> 500	> 60%



a) Convectional process (heave computation and slow response)

b) Pre-computing approach (computation distributed, rapid response)

Figure 2. Using pre-computing to enable rapid response to fault localization requests in the CI environment [11]

Our observations suggest that techniques are typically grouped into two categories. CBFL techniques (Group I) like Tarantula are time-saving but of relatively coarse accuracy. The utilization of these techniques simply depends on the program spectra and testing results, and the information can be easily acquired dynamically from software testing. They need less than five seconds on average in running time to produce a fault localization result, and can locate less than 30% of all faults when examining up to 5% of the code. In detail, the time cost of Tarantula is 0.04 seconds, and the measured accuracy is 47.15%, proportion of localized faults. On the other hand, the program state-based techniques (Group II) such as mutation are more accurate in locating faults but are also more time-consuming. In detail, the running time of [7] is 40.02 minutes in average and the measured accuracy is 60.16%.

It is natural to think about whether we can synthesize fault localization techniques that are both effective and efficient. Since high accuracy comes from sufficient computation. We notice that fault localization starts when a testing request is triggered. To wait for a heavy computation to

finish, the response rate becomes too low to accept, for techniques like [7]. We therefore want to use pre-computing to provide rapid response rate, so that both high accuracy and seemingly low-cost can be achieved.

Figure 2 shows our basic idea. Plot (a) shows the original sequence process of the conventional testing paradigm in a continuous integration environment. The program evolves from a long coding phase and finally converges to a stable version. During testing, the program is run against an equipped test suite to collect the coverage information and test results. When failed runs appear, faults in the program are confirmed. Coverage information is input to a testing server running a fault localization service. After a complicated and long computing process, an accurate fault localization result is generated.

From plot (a), we observe that in the coding phase, testing server is always idle, and at the moment a testing request is triggered, the testing server is fully occupied and the programmer is waiting. If we could distribute the computation into idle period, the waiting time for a programmer to collect fault localization results can be shortened. Plot (b) of Figure 2 illustrates the idea. During coding, we want to adopt the pre-computing strategy to generate intermediate data, so that the testing server is reasonably used. At the moment a testing request is triggered, a fault localization result can be rapidly computed using the pre-computing results. Thus, the cost of adapted fault localization technique is reduced, without loss of its accuracy.

The proposed idea seems interesting and workable. However, we realize that pre-computing localization results in the coding phase means that locating faults for a future program P. The program P may be not available in the pre-computing process. On the other hand, there will be continuous program versions P' during the coding period and they can be used to provide useful information for the program P. As a result, the pre-computing process can only be carried on P', and at least two issues exist.

1. Since pre-computing is needed, what fault localization technique can be conducted on P' instead of P?

2. How can we make up for the locating accuracy loss caused by using P' instead of P in the pre-computing?

4. RESEARCH EXPLORATION

A. Fault localization in a CI environment

In the continuous integration (CI) [2] environment, program versions are continuously updated, and testing is requested at each commit, which forms a coding-testing-debugging loop. High-accuracy fault localization techniques are mostly time-consuming and cannot be done efficiently. It is deemed a bottleneck. It can be extremely significant to increase the response rate of such techniques.

B. Selecting promising techniques

Let us recall the two issues raised in Section III. We are aiming at localizing faults of the program P. Nevertheless, we need the testing server to pre-compute the information while the code is an

intermediate version, say P'. Therefore, in our method, we often get the information from an in-complete and previous program version. However, getting the accurate running results for program version P is essential for the conventional CBFL technique. For instance, Tarantula uses the proportions of passed and failed executions to compute the suspiciousness of each statement in a program. The passed and failed executions must be from an actual program. Obviously, these CBFL techniques are not suitable for pre-computing.

In the next section, we will disclose that there exists at least one fault localization technique, which does not fully rely on the program P to produce fault localization results. Further, we will show how to conduct pre-computing with it and how to adapt it to a CI environment to synthesize a novel testing paradigm.

C. Mutation-enabled fault localization

Mutation testing is proposed to evaluate the quality of a program and measure the failure-revealing capacity of a test suite by manipulating a program using mutation operators.

Some research endeavors have explored mutation analysis to provide accurate fault-localization. Papadakis et al. [7] proposed a mutation-based fault localization technique. Basically, a mutation version, which corresponding test result is very similar to that of the original program, is deemed that the mutant statement is close to the fault position. Formally, given a program P and the associated test suite T. Mutant versions such as P1, P2 and so on are generated using mutation operators. The distinction between P and Pi (i=1, 2, …) can be decided by comparing the program outputs of them, say P(T) and Pi(T). If the output of some mutation version is very similar to that of P, the embedded mutant is more likely to be located on the faulty program statement.

Since most of the computation cost is due to executing mutation versions Pi, we need to pre-compute them. We design our methodologies with the following considerations.

1. During development, modification region of the program is often centralized. Therefore, the complete version P and intermediate version P' are close to each other.

2. According to above, we can generate mutation versions P1, P2, … from the intermediate version P' instead of from P at the coding phase.

We thus enable the mutation-based technique [7] in a pre-computing paradigm, and propose a four-step process, illustrated in Figure 3.

[Step 1]: Generating the mutation versions P1, P2, … from the intermediate version P'.

[Step 2]: Executing the mutation versions P1, P2, … and collect their execution results O1, O2, …., respectively.

[Step 3]: When coding finishes and a testing request is triggered, executing the stable version P to get result O.

[Step 4]: Use the distance between Oi and O to predict the distance between faults with the mutated statement in Pi.

D. Visions and challenges

We propose to use pre-computing to speed up a fault localization technique, which can rapidly respond testing request in a CI environment. Further, we apply the method on existing high-accurate techniques (e.g., [7]), which makes our approach to have adequate accuracy.

Challenges mainly relate to threats to the effectiveness of our approach and the preliminary solution of the problem, in the following aspects.

1. It is recognized in the software engineering research community that mutants may not couple well with real faults. We will integrate high-order mutants to simulate real faults.

2. Previous work indicates that better localization results are achieved by large number of mutation versions [7]. Our approach can be extended by increasing computing resources, since the executing of mutation versions can be in parallel.

3. The scale of program can be very large and seriously increases the complexity. We can focus on the critical region or modifying region to carry out mutations.

5. CONCLUSION

Accurate fault localization is achieved by high-cost computation. In this paper, we propose to make use of pre-computing to distribute time-intensive computation to idle period of coding phase. Our approach consists of two steps, to find fault localization techniques that can be pre-computed, and to adapt it to the pre-computing paradigm in a continuous integration environment. We demonstrate our exploration using an existing mutation-based fault localization technique, and thus found a fault localization technique both effective and efficient. We are now developing our method gradually. Future work includes development of mechanisms integrating other fault localization techniques like value replacement.

ACKNOWLEDGEMENTS

Thanks to Zhenyu Zhang and Jeffrey Hu for their help on sharpening the idea.

REFERENCES

[1] R. Abreu, P. Zoeteweij, and A. J C Van Gemund. On the accuracy of spectrum-based fault localization. In Testing: Academic and Industrial Conference Practice and Research Techniques, pages 89-98, 2007.

[2] B. Jiang, and T. Y. Chen. How well do test case prioritization techniques support statistical fault localization?. In Computer Software and Applications Conference, pages 99-106, 2009.

[3] D. Jeffrey, N. Gupta, and R. Gupta. Fault localization using value replacement. In Proceedings of the 2008 International Symposium on Software Testing and Analysis, pages 167-178, 2008.

[4] B. Liblit, M. Naik, A. X. Zheng, A. Aiken, and M. I. Jordan. Scalable statistical bug isolation. In Proceedings of the 2005 ACM SIGPLAN Conference on Programing Language Design and Implementation, pages 15-26, 2005.

[5] C. Liu, L. Fei, X. Yan, J. Han, and S. P. Midkiff. Statitical debugging: A hypothesis testing-based approach. IEEE Transactions on Software Engineering, 2006.

[6] L. Naish, H. J. Lee, and L. Ramamohanarao. A model for spectra-based software diagnosis. ACM Transactions on Software Engineering Methodology, pages 20-23, 2011.

[7] M. Papadakis and Y. Le-Traon. Using Mutants to Locate "Unknown" Faults. Software Testing, Verification and Validation, pages 691-700, 2012.

[8] X. Xie, T. Chen, F. C. Kuo, and B. Xu. A theoretical analysis of the risk evaluation formulas for spectrum-based fault localization. ACM Transactions on Software Engineering and Methodology, 2013

[9] Y. Yu, J. A Jones, and M. J. Harrold. An empirical study of the effects of test-suite reduction on fault localization. In the 30th International Confference on Software Engineearin, 2008.

[10] X. Zhang, and R. Gupta. Locating faults through automated predicate switching. In proceedings of the 28th International Conference on Software Engineering, pages 272-281, 2006.

[11] Z. Zhang, H. Li, A Technical Report to Continuously Predict the Dynamic Behaviors of Evolving Programs under Development, State Key Laboratory of Computer Science. Technical reports ISCAS-SKLCS-14-11.

TESTING AND IMPROVING LOCAL ADAPTIVE IMPORTANCE SAMPLING IN LJF LOCAL-JT IN MULTIPLY SECTIONED BAYESIAN NETWORKS

Dan Wu[1] and Sonia Bhatti[2]

[1]School of Computer Science University of Windsor, Windsor, Ontario Canada
danwu@uwindsor.ca
[2]School of Computer Science University of Windsor, Windsor, Ontario Canada
bhattif@uwindsor.ca

ABSTRACT

Multiply Sectioned Bayesian Network (MSBN) provides a model for probabilistic reasoning in multi-agent systems. The exact inference is costly and difficult to be applied in the context of MSBNs as the size of problem domain becomes larger and complex. So the approximate techniques are used as an alternative in such cases. Recently, for reasoning in MSBNs, LJF-based Local Adaptive Importance Sampler (LLAIS) has been developed for approximate reasoning in MSBNs. However, the prototype of LLAIS is tested only on Alarm Network (37 nodes). But further testing on larger networks has not been reported yet, so the scalability and reliability of algorithm remains questionable. Hence, we tested LLAIS on three large networks (treated as local JTs) namely Hailfinder (56 nodes), Win95pts (76 nodes) and PathFinder(109 nodes). From the experiments done, it is seen that LLAIS without parameters tuned shows good convergence for Hailfinder and Win95pts but not for Pathfinder network. Further when these parameters are tuned the algorithm shows considerable improvement in its accuracy and convergence for all the three networks tested.

KEYWORDS

MSBN, LJF, Adaptive Importance sampling, Tunable parameters

1. INTRODUCTION

Multiply Sectioned Bayesian Networks (MSBN) is the model grounded on the idea of cooperative multi-agent probabilistic reasoning, is an extension of the traditional Bayesian Network model and it provide us with solution to the probabilistic Reasoning under cooperative agents. The Multiple agents [1] collectively and cooperatively reason about their respective problem domain on the basis of their local knowledge, local observation and limited inter-agent communication. Typically the inference in MSBN is generally carried out in some secondary structure known as *linked Junction tree forest* (LJF). The LJF provides a coherent framework for exact inference with MSBN [2], LJF constitutes local Junction trees (JT) and linkage trees for making connections between the neighbouring agents to communicate among themselves. Agents communicate through the messages passed over the LJF linkage trees and belief updates in each LJF local junction tree (JT) are performed upon the arrival of a new inter-agent message.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 67–78, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60606

However the computational cost of exact inference makes it impractical for larger and complex domains. So the approximate inference algorithms are being used to estimate the posterior beliefs. Hence, it is very important to study the practicability and convergence properties of sampling algorithms on large Bayesian networks.

To date there are many stochastic sampling algorithms proposed for Bayesian Networks and are widely used in BN approximation but this area is quite problematic, since many attempts have been made in developing MSBN approximation algorithms but all of these forgo the LJF structure and sample MSBN directly in global context. Also it has been shown that such type of approximation requires more inter-agent message passing and also leaks the privacy of local subnet [3]. So, sampling MSBN in global context is not good idea as it analyses only small part of entire multi-agent domain space. So in order to examine local approximation and to maintain LJF framework, the sampling process is to be done at each agent's subnet. The LJF-based Local adaptive Importance Sampler (LLAIS) [3] is an example of extension of BN Importance sampling techniques to JT's. An important aspect of this algorithm is that it facilitates inter-agent message calculation along with the approximation of the posterior probabilities.

So far the application of LLAIS is done on smaller network consisting of 37 nodes which is treated as local JT in LJF. LLAIS produced good estimates of local posterior beliefs for this smaller network but its further testing on larger sizes of local JTs is not reported yet. We tested LLAIS for its scalability and reliability on the three larger networks treating them as local JTs in LJF. It is important to test the algorithm since the size of local JT can vary and can go beyond 37 nodes network, on which preliminary testing has been done. Our testing demonstrated that without tuning of parameters, LLAIS is quite scalable for Hailfinder (56 nodes) and Win95pts (76 nodes) but once it is applied to Pathfinder (109 nodes) network its performance deteriorates. Further, when these parameters are tuned properly it resulted in significant improvement in the performance of algorithm, now it requires less number of samples and less updates than required by the original algorithm to give better results.

2. BACKGROUND

2.1 Multiply Sectioned Bayesian Networks (MSBNs)

In this paper, we assume that the reader is familiar with Bayesian networks (BNs) and basic probability theory [4]. The Multiply Sectioned Bayesian Networks (MSBNs) [2] extend the traditional BN model from a single agent oriented paradigm to the distributed multi-agent paradigm and provides a framework to apply probabilistic inference in distributed multi-agent systems. Under MSBNs, a large domain can be modelled modularly and the inference task can be performed in coherent and distributed fashion.

The **MSBN model** is based on the following five assumptions:

1. Agent's belief is represented as probability.

2. Agents communicate their beliefs based on a small set of shared variables.

3. A simpler agent organization is preferred.

4. A DAG is used to structure each agent's knowledge.

5. An agent's local JPD admits the agent's belief of its local variables and the shared variables with other agents.

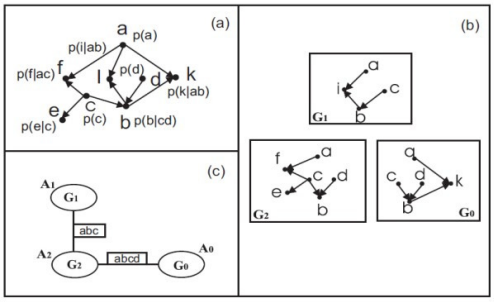


Figure 1: (a) A BN (b) A small MSBN with three subnets (c) the corresponding MSBN hypertree.

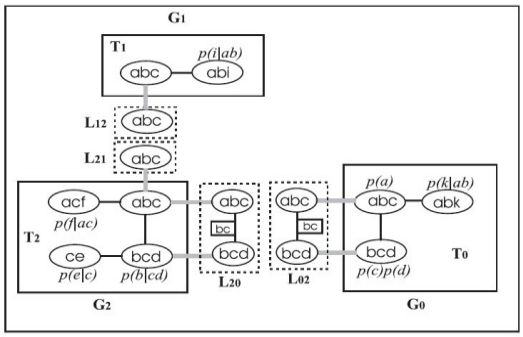


Figure 2. An MSBN LJF shown with initial potentials assigned to all the three subnets.

MSBN consist of set of BN subnets where each subnet represents the partial view of a larger problem domain. The union of all subnet DAGs must also be DAG, denoted by G. These subnets are organised into a tree structure called a *hypertree* [2] denoted by ψ. Each hypertree node, known as *hypernode*, corresponds to a subnet; each hypertree link, known as *hyperlink*, corresponds to a *d-sepset*, which is set of shared variables between the adjacent subnets. A hypertree ψ is purposely structured so that (1) for any variable x contained in more than one subnet with its parents $\pi(x)$ in G, there must exist a subnet containing $\pi(x)$; (2) shared variables between two subnets N_i and N_j are contained in each subnet on the path between N_i and N_j in ψ. A hyperlink renders two sides of the network conditionally independent similar to the separator in a junction tree (JT).

Fig. 1 (a) shows BN which is sectioned into MSBN with three subnets in Fig. 1(b) and Fig. 1(c) shows the corresponding *hypertree* structure. A derived secondary structure called *linked junction tree forest*(LJF) is used for inference in MSBNs; it is constructed through a process of cooperative and distributed compilation where each *hypernode* in *hypertree* ψ is transformed into local JT, and each hyperlink is transformed into a *linkage tree*, which is a JT constructed from d-sepset. Each cluster of a linkage tree is called a *linkage*, and each separator, a *linkage separator*. The cluster in a local JT that contains a linkage is called a *linkage host*. Fig. 2 shows the LJF constructed from the MSBN in Fig.1 (b) and (c). Local JTs, T_0, T_1 and T_2 are constructed from BN subnets G_0, G_1 and G_2 respectively, are enclosed by boxes with solid edges. The linkage trees; $L_{20}(L_{02})$ and $L_{21}(L_{12})$, are enclosed by boxes with dotted edges. The linkage tree L_{20} contains two linkages $\{a,b,c\}$ and $\{b,c,d\}$ with linkage separator bc (not shown in the figure). The linkage hosts of T_0 for L_{02} are clusters $\{a,b,c\}$ and $\{b,c,d\}$.

3. BASIC IMPORTANCE SAMPLING FOR LJF

Here we assume that readers are aware of basic importance sampling for LJF local JT. The research done so far has highlighted the difficulties in applying stochastic sampling to MSBNs at a global level [5]. Direct local sampling is also not feasible due to the absence of a valid BN structure [3]. However, an LJF local JT can be calibrated with a marginal over all the variables [6] making local sampling possible. Algorithms proposed earlier combine sampling with JT belief propagation but do not support efficient inter-agent message calculations in context of MSBNs.

The [3] introduced a JT-based importance sampler by defining an explicit form of the importance function so that it facilitates the learning of the optimal importance function. The JPD over all the variables in a calibrated local JT can be obtained similar to Bayesian network DAG factorization.

Let $C_1, C_2 \text{K K}, C_m$ be the m JT clusters given in the ordering which satisfies the running intersection property. The separator $S_i = \varnothing$ for $i = 1$ and $S_i = C_i \cap (C_1 \cup C_2 \cup \text{K K} \cup C_{i-1})$ for $i = 2,3,\text{K}, m$. Since $S_i \subset C_i$, the residuals are defined as $R_i = C_i \setminus S_i$. The junction tree running intersection property guarantees that the separator S_i separates the residual R_i from the set $(C_1 \cup C_2 \cup \text{K K} \cup C_{i-1}) \setminus S_i$ in JT.

Thus applying the chain rule to partition the residues given by the separators and have JPD expressed as $P(C_1, \text{K K}, C_m) = \prod_{i=1}^{m} P(R_i \mid S_i)$. The main idea is to select the root from the JT clusters and then directing all the separators away from the root forming a directed sampling JT. It is analogous to BN since both follow recursive form of factorization.

Once the JPD has been defined for LJF local JT, the importance function P' in basic sampler is defined as:

$$P'(X \setminus E) = \prod_{i=1}^{m} P(R_i \setminus E \mid S_i) \mid_{E=e} \qquad (1)$$

The vertical bar in $P(R_i \setminus E \mid S_i) \mid_{E=e}$ indicates the substitution of **e** for **E** in $P(R_i \setminus E \mid S_i)$. This importance function is factored into set of local components each corresponding to the JT clusters. It means when the calibrated potential is given on each JT cluster C_i we can easily compute for every cluster the value of $P(R_i \mid S_i)$ directly. For the root cluster: $P(R_i \mid S_i) = P(R_i) = P(C_i), i = 0$.

We traverse a sampling JT and sample variables of the residue set in each cluster corresponding to the local conditional distribution. This sampling is similar to the BN sampling except now group of nodes are being sampled and not the individual nodes. Whenever cluster is encountered with the node in the evidence set **E**, it will be assigned value which is given by evidence assignment. A complete sample consist of the assignment to all the non- evidence nodes according to the local JT's prior distribution.

The score for each sample can be computed as:

$$Score_i = \frac{P(S_i, E)}{P'(S_i)} \qquad (2)$$

The score so computed in Equation 2 will be used in LLAIS algorithm for adaptive importance sampling. It is proven that the optimal importance function for BN importance sampling is the posterior distribution $P(X \mid E = e)$ [7]. Applying this result to JTs, we can define the optimal importance function as:

$$\rho(X \setminus E) = \prod_{i=1}^{m} P(R_i \setminus E \mid E = e) \qquad (3)$$

The above Equation 3 takes into account the influence of all the evidences from all clusters in the sample of current cluster.

3.1 LJF-Based Local Adaptive Importance Sampler (LLAIS)

In 2010, LJF local JT importance sampler called LLAIS [3] was designed that follows the principle of adaptive importance sampling for learning factors of importance function. This algorithm was specifically proposed for the approximation of posteriors in case of local JT in LJF providing the framework for calculation of inter-agent messages between the adjacent local JTs.

The sub-optimal importance function used for LJF Local Adaptive Importance Sampling is as follows,

$$\rho(X \setminus E) = \prod_{i=1}^{m} P(R_i \setminus E \mid S_i, E = e) \qquad (4)$$

This importance function is represented in the form of set of local tables. This importance function is learned to approach the optimal sampling distribution.

These local tables are called the ***Clustered Importance Conditional Probability Table*** **(CICPT).** These CICPT tables are created for each local JT cluster consisting of the probabilities indexed by the separator to the precedent cluster (based on the cluster ordering in the sampling tree) and conditioned by the evidence.

For non-root JT clusters, CICPT table are defined in the form of $P(R_i \mid S_i, E)$, and for the JT root cluster, CICPT table are of the form of $P(R_i \mid S_i, E) = P(C_i \mid E)$.

The learning strategy is to learn these CICPT tables on the basis of most recent batch of samples and hence the influence of all evidences is counted through the current sample set. These CICPT tables have the structure similar to the factored importance function and are alike to an ICPT table of Adaptive Importance Sampling of BN in the previous section 4.1 and are updated periodically by the scores of samples generated from the previous tables.

Algorithm for LLAIS

Step 1. Specify the total number of samples M, total updates K and update interval L, Initialize the CICPT tables as in Equation 4.

Step 2. Generate L samples with the scores according to the current CICPT tables. Estimate $P'(R_i \mid S_i, e)$ by normalizing the scores for each residue set given the states of separator set.

Step 3. Update the CICPT tables based on the following learning function [45]:

$$P^{K+1}(R_i \mid S_i, e) = (1-\eta(k))P^k(R_i \mid S_i, e) + \eta(k)P'(R_i \mid S_i, e),$$

where $\eta(k)$ is the learning rate.

Step 4. Modify the importance function if necessary, with the heuristic of Є-cutoff. For the next update, go to Step 2.

Step 5. Generate the samples from the learned importance function and calculate scores as in Equation 2.

Step 6. Output the posterior distribution for each node.

In LLAIS the importance function is dynamically tuned from the initial prior distribution and samples obtained from the current importance function are used to refine gradually the sampling distribution. It is well known that thick tails are desirable for importance sampling in BNs. The reason behind it is that the quality of approximation deteriorates in the presence of probabilities due to generation of large number of samples having zero weights [3]. This issue is solved using the heuristic Є-cutoff [7], the small probabilities are replaced with Є if less than a threshold Є, and the change is compensated by subtracting the difference from the largest probability.

4. IMPROVING LLAIS BY TUNING THE TUNEABLE PARAMETERS

The tuneable parameters plays vital role in the performance of sampling algorithm. There are many tuneable parameters in LLAIS such as the heuristic value of threshold $\in$-cutoff, updating intervals, number of updates, number of samples and learning rate discussed as follows:

1. Threshold $\in$-cutoff – it is used for handling very small probabilities in the network. The proper tuning helps the tail of importance function not to decay faster, the optimal value for $\in$-cutoff is dependent upon the network and plays key role in getting better precision these experiments with different cut-off values are motivated from [8].

2. Number of updates and updating interval - the number of updates plays an important role in the sense that it denotes how many times the CICPT table has to be updated so that it will result in optimal output and updating interval denotes the number of samples that have to be updated.

3. Number of samples - plays very important role in the stochastic sampling algorithm as the performance of sampling increases with the number of samples. It is always good to have minimum number of samples that can help you reach better output for it will be time and cost efficient

4. Learning Rate - in [7] is defined as the rate at which optimal importance function will be learned as per the formula $\eta(k) = a(\frac{a}{b})^{k/k\max}$, where a = initial learning rate, b = learning rate in the last step, k = number of updates and k max = total number of updates.

These tuneable parameters are tuned after many experiments in which they were given heuristically different values and then checked for performance. Table 1 shows the comparison of values of various tuneable parameters for original and improved LLAIS.

Table 1: Shows the comparison of values of various tuneable parameters for original LLAIS and improved LLAIS.

Tunable parameters	Original LLAIS	Improved LLAIS
Number of samples	5000	4500
Number of updates	5	3
Updating interval	2000	2100
Threshold value	Nodes with outcomes <5	Nodes with outcomes < 5
	0.05	0.01
	Nodes with outcomes < 8	Nodes with outcomes < 8
	0.005	0.006
	Else = 0.0005	Else = 0.0005

5. EXPERIMENT RESULTS

We used Kevin Murphy's Bayesian Network toolbox in MATLAB for experimenting with LLAIS. For testing of LLAIS algorithm, the exact importance function is computed, which is considered to be the optimal one and then its performance of sampling is compared with that of approximate importance function in LLAIS. The testing is done on Hailfinder (56 nodes), Win95pts (76 nodes) and Pathfinder (109 nodes), which are treated as local JT in LJF. The approximation accuracy is measured in terms of *Hellinger's distance* which is considered to be perfect in handling zero probabilities which are common in case of BN.

From [8], The Hellinger's distance between two distributions F_1 and F_2 which have the probabilities $P_1(x_{ij})$ and $P_2(x_{ij})$ for state $j(j = 1,2,\text{K K},n_i)$ of node i respectively, such that $X_i \notin E$ is defined as:

$$H(F_1, F_2) = \sqrt{\frac{\sum_{X_i \in N \setminus E} \sum_{j=1}^{n_i} \{\sqrt{P_1(x_{ij})} - \sqrt{P_2(x_{ij})}\}^2}{\sum_{X_i \in N \setminus E} n_i}} \qquad (5)$$

where N is the set of all nodes in the network, E is the set of evidence nodes and n_i is the number of states for node i. $P_1(x_{ij})$ and $P_2(x_{ij})$ are sampled and exact marginal probability of state j of node i.

5.1 Experiment Results for Testing LLAIS

For each of the three networks we generated in total 30 test cases consisting of the three sequences of 10 test cases each. The three sequences include 9, 11 and 13 evidence nodes respectively. For each of the three networks, LLAIS with exact and approximate importance function is evaluated using $M = 5000 samples$. With LLAIS using approximate importance function, the learning function used is $\eta(k) = a(\frac{a}{b})^{k/k\max}$ and set $a = 0.4$ and $b = 0.14$, total updates $K = 5$ and each updating step, $L = 2000$. The exact importance function is optimal hence it does not require updating and learning.

Fig.4 shows the results for all the 30 test cases generated for Hailfinder network. Each test case was run for 10 times and average *Hellinger's distance* was recorded as a function of $P(E)$ to measure the performance of LLAIS as $P(E)$ goes more and more unlikely. It can be seen that LLAIS using approximate importance function performs quite well and shows good scalability for this network.

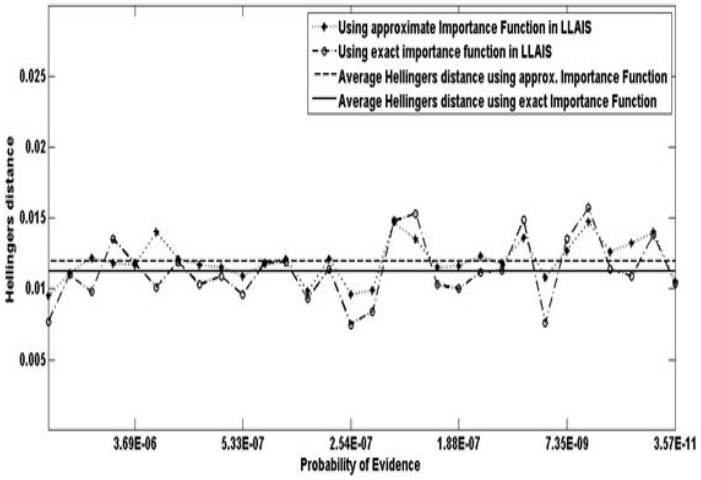


Figure 4: Performance comparison of approximate and exact importance function combining all the 30 test cases generated in terms of Hellinger's distance for Hailfinder network.

Fig. 5 shows the results generated for all the 30 test cases generated from Win95pts network. It can be concluded that for this network too LLAIS using approximate importance function shows good scalability and its performance is quite comparable with that using exact importance function.

Fig. 6 shows the results generated for all the 30 test cases generated from Pathfinder networkIt is seen that for this network LLAIS performed poor, the reason is the presence of extreme probabilities which needs to deal with. Hence LLAIS doesn't prove to be scalable and reliable for this network.

Table 2 below shows the comparison of the statistical results for all the 30 test cases generated using approximate and exact importance function in LLAIS.

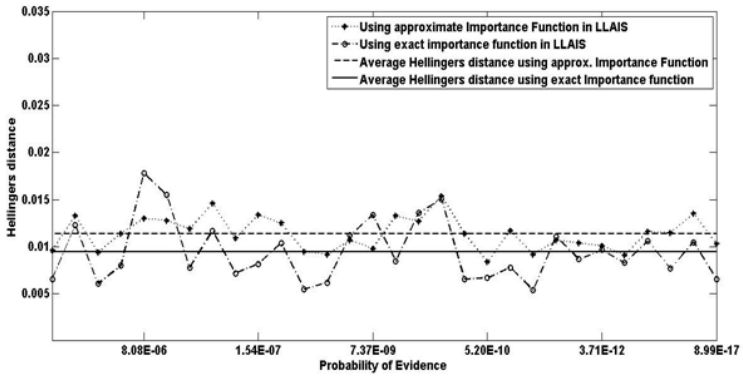


Figure 5: Performance comparison of approximate and exact importance function combining all the 30 test cases generated in terms of Hellinger's distance for Win95pts network.

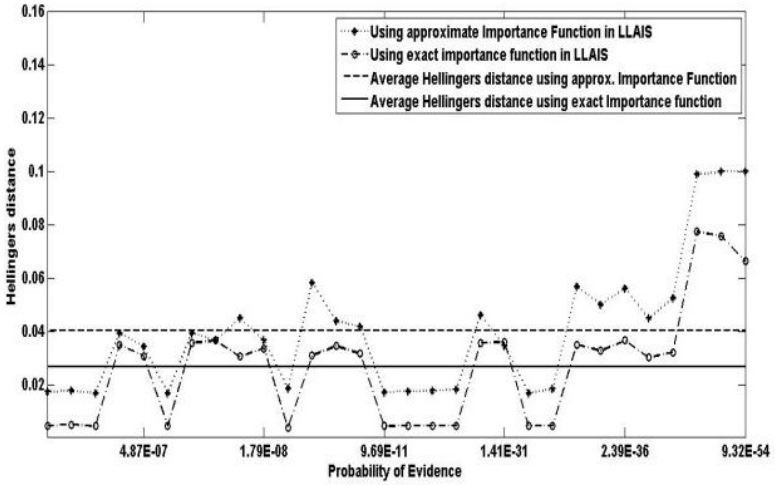


Figure 6: Performance comparison of approximate and exact importance function combining all the 30 test cases generated in terms of Hellinger's distance for Pathfinder network.

Table 2: Comparing the statistical results for all 30 test cases generated for testing LLAIS for all the three networks.

Name of ne	**Hailfinder network**	
Hellinger's	**Approx. imp**	**Exact imp**
Minimum Error	0.0095	0.0075
Maximum Error	0.0147	0.0157
Mean	0.0118	0.0113
Median	0.0118	0.0111
Variance	1.99E-06	4.92E-06
Name of ne	**Win95pts network**	
Hellinger's	**Approx. imp**	**Exact imp**
Minimum Error	0.0084	0.0054
Maximum Error	0.0154	0.0178
Mean	0.0114	0.0095
Median	0.0114	0.0084
Variance	3.18E-06	1.03E-05
Name of ne	**Pathfinder network**	
Hellinger's	**Approx. imp**	**Exact imp**
Minimum Error	0.0168	0.0038
Maximum Error	0.1	0.0774
Mean	0.0403	0.0269
Median	0.0379	0.0313
Variance	6.05E-04	4.41E-04

5.2 Experiment Results for Improved LLAIS

After tuning the parameters as discussed in section 4, LLAIS shows considerable improvement in its accuracy and scalability with proper tuning of tunable parameters. Now the Improved LLAIS uses less number of samples and less updates in comparison to the Original LLAIS for giving posterior beliefs.

Fig. 7 shows the comparison of performance of Original LLAIS with Improved LLAIS and it can be seen that Improved LLAIS performs quite well showing good scalability on Hailfinder network.

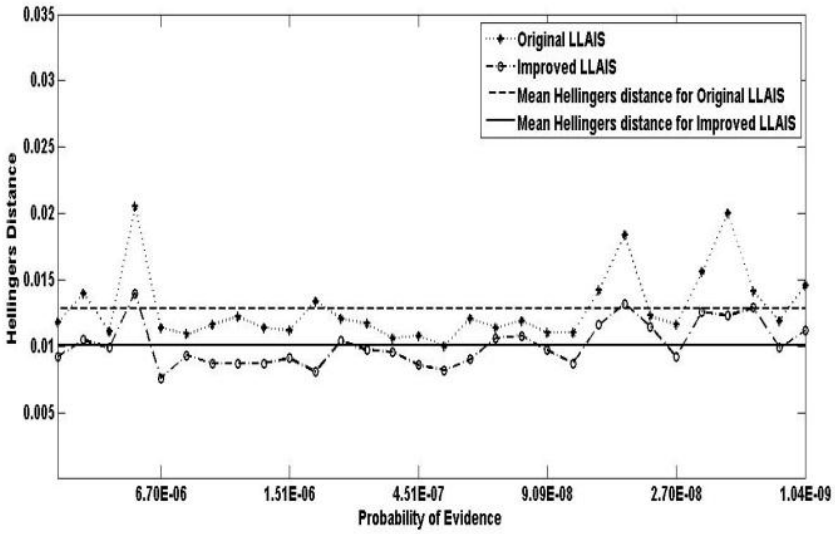


Figure 7: Performance comparison of Original LLAIS and Improved LLAIS for Hailfinder network. Hellinger's distance

Fig. 8 shows the comparison of performance of Original LLAIS with Improved LLAIS for Win95pts network and it can be seen in the graph that here also Improved LLAIS performed quite well with less errors as compared to the Original LLAIS.

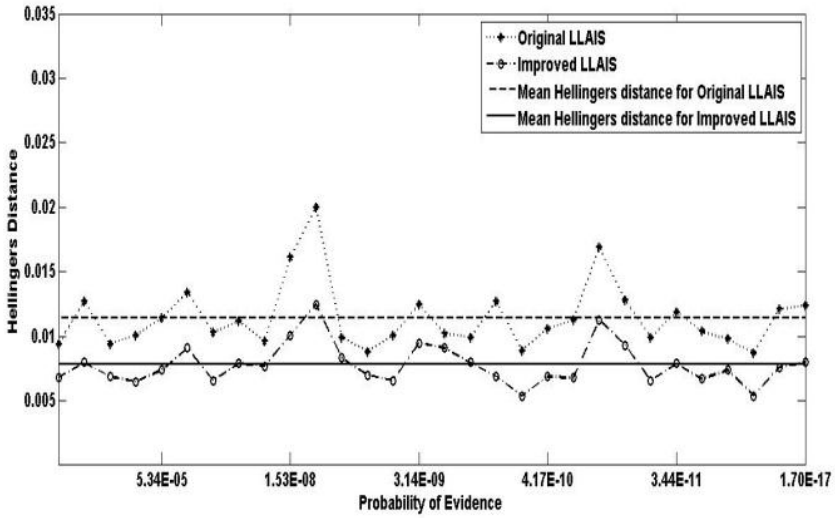


Figure 8: Performance comparison of Original LLAIS and Improved LLAIS for Win95pts network. Hellinger's distance for each of the 30 test cases plotted against $P(E)$

Fig 9 shows the comparison of performance of Improved LLAIS with Original LLAIS. The most extreme probabilities are found in this network, hence adjustments with threshold values played a key role in improving the performance; hence after tuning the parameters Improved LLAIS showed better performance in comparison to the original one for this network.

Table 3 shows the comparison of statistical results from all 30 test cases generated for Improved LLAIS and Original LLAIS.

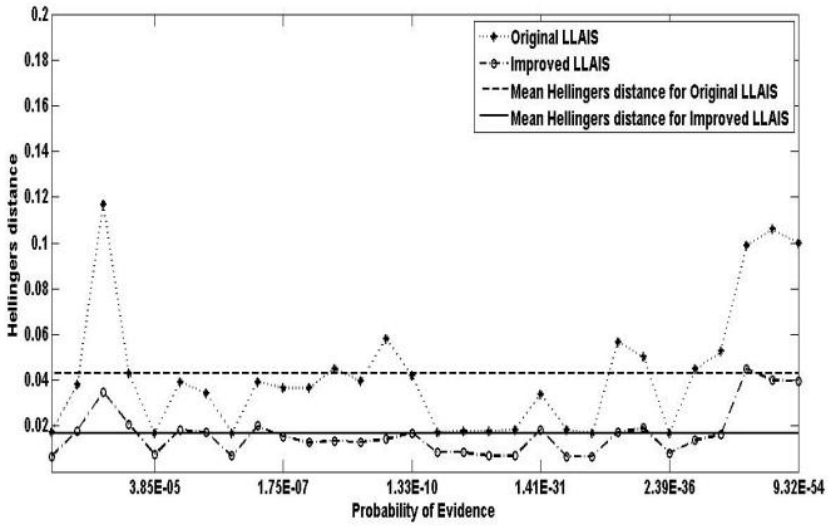


Figure 9: Performance comparison of original LLAIS and improved LLAIS for Pathfinder network. Hellinger's distance for each of the 30 test cases plotted against $P(E)$

Table 3: shows the comparison of results for original LLAIS and Improved LLAIS from all 30 test cases generated.

Name of ne	**Hailfinde r ne twork**	
Hellinger's	**Orignal**	**Improve d**
Minimum Error	0.01	0.0076
Maximum Error	0.0205	0.014
Mean	0.0128	0.0101
Median	0.0119	0.0097
Variance	7.08E-06	2.73E-06
Name of ne	**Win95pts ne twork**	
Hellinger's	**Orignal**	**Improve d**
Minimum Error	0.0087	0.0054
Maximum Error	0.02	0.0125
Mean	0.0114	0.0078
Median	0.0105	0.0075
Variance	6.45E-06	2.50E-06
Name of ne	**Pathfinde r ne twork**	
Hellinger's	**Orignal**	**Improve d**
Minimum Error	0.0168	0.0068
Maximum Error	0.117	0.0451
Mean	0.0427	0.0166
Median	0.0387	0.0149
Variance	7.80E-04	1.09E-04

6. CONCLUSION AND FUTURE WORKS

LLAIS is the extension of BN importance sampling to JTs. Since the preliminary testing of the algorithm was done only on smaller local-JT in LJF of 37 nodes, hence the scalability and reliability of the algorithm was questionable as the size of local-JTs may vary. From the experiments done, it can be concluded that LLAIS without parameters tuned performs quite well on local-JT of size 56 and 76 nodes but its performance deteriorates on 109 nodes network due to presence of extreme probabilities, once the parameters are tuned algorithm shows considerable improvement in its accuracy. It has been seen that learning time of the optimal importance function takes too long, so the choice of initial importance function $\Pr^0(X \setminus E)$ close to the

optimal importance function can greatly affect the accuracy and convergence in the algorithm. As mentioned in [3], there is still one important question that remains unanswered how the local accuracy will affect the overall performance of the entire network. Further experiments are still to be done on the full scale MSBNs.

REFERENCES

[1] Karen H. Jin, ―Efficient probabilistic inference algorithms for cooperative Multi-agent Systems‖, Ph.D. dissertation, University of Windsor (Canada), 2010.

[2] Y.Xiang, Probabilistic Reasoning in Multiagent Systems: A Graphical Models Approach‖. Cambridge University Press, 2002.

[3] Karen H. Jin and Dan Wu, ―Local Importance Sampling in Multiply Sectioned Bayesian Networks‖, Florida Artificial Intelligence Research Society Conference, North America, May. 2010.

[4] Daphne Koller and Nir Friedman, Probabilistic Graphical Models-Principles and Techniques, MIT Press, 2009.

[5] Y.Xiang, ―Comparison of multiagent inference methods in Multiply Sectioned Bayesian Networks‖, International journal of approximate reasoning, vol. 33, pp.235-254, 2003.

[6] K.H.Jin and D.Wu, ―Marginal calibration in multi-agent probabilistic systems‖, In Proceedings of the 20th IEEE International conference on Tools with AI, 2008.

[7] J. Cheng and M. J. Druzdzel, ―BN-AIS: An adaptive importance sampling algorithm for evidential reasoning in large Bayesian networks‖, Artificial Intelligence Research, vol.13, pp.155–188, 2000.

[8] C. Yuan, ―Importance Sampling for Bayesian Networks: Principles, Algorithms, and Performance‖, Ph.D. dissertation, University of Pittsburgh, 2006.

WEIGHTS STAGNATION IN DYNAMIC LOCAL SEARCH FOR SAT

Abdelraouf Ishtaiwi

Faculty of Information Technology, University of Petra, Amman, Jordan
faishtaiwig@uop.edu.jo

ABSTRACT

Since 1991, tries were made to enhance the stochastic local search techniques (SLS). Some researchers turned their focus on studying the structure of the propositional satisfiability problems (SAT) to better understand their complexity in order to come up with better algorithms. Other researchers focused in investigating new ways to develop heuristics that alter the search space based on some information gathered prior to or during the search process. Thus, many heuristics, enhancements and developments were introduced to improve SLS techniques performance during the last three decades. As a result a group of heuristics were introduced namely Dynamic Local Search (DLS) that could outperform the systematic search techniques. Interestingly, a common characteristic of DLS heuristics is that they all depend on the use of weights during searching for satisfiable formulas.

In our study we experimentally investigated the weights behaviors and movements during searching for satisfiability using DLS techniques, for simplicity, DDFW DLS heuristic is chosen. As a results of our studies we discovered that while solving hard SAT problems such as blocks world and graph coloring problems, weights stagnation occur in many areas within the search space. We conclude that weights stagnation occurrence is highly related to the level of the problem density, complexity and connectivity.

1. INTRODUCTION

The propositional satisfiability (SAT) problem is at the core of many computer science and artificial intelligence problems. Hence, finding efficient solutions for SAT has far reaching implications. In this study, we consider propositional formulae in conjunctive normal form (CNF): $\mathcal{F} = \bigwedge_m \bigvee_n l_{mn}$ in which each l_{mn} is a literal (propositional variable or its negation), and each disjunct $\bigvee_n l_{mn}$ is a clause. The problem is to find an assignment that satisfies $\mathcal{F}$. Given that SAT is NP complete, systematic search methods can only solve problems of limited size. On the other hand, relatively simple stochastic local search (SLS) methods have proved successful on a wide range of larger and more challenging problems [10]. Furthermore, stochastic local search (SLS) techniques are proven to be effective in solving hard satisfiability boolean problems.

However, their performance is still arguably poor when compared to systematic search techniques. Therefore, and since the development of the first clause weighting dynamic local search (DLS) algorithms for SAT, namely he Breakout heuristic [16], tries were made to enhance the local search techniques in many different ways. Some researchers turned their focus on studying the structure of the satisfiability problems (as in 1.1) to better understand the complexity

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 79–90, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60607

of it and to come up with algorithms that could solve the problems in an optimal way. Other researchers focused on investigating new heuristics that alter the search space based on some information gathered during searching for a solution (as in 1.2). Thus, many heuristics, enhancements and developments were introduced to improve SLS techniques performance in the last three decades. As a result, a group of heuristics were introduced that could outperformed the systematic search techniques, namely Dynamic Local Search (DLS). Recent DLS algorithms depend on the use of weights to alter the search space. In other words, weights are used when there are no moves that could decrease the search cost, to make it possible for the technique to take unattractive moves which could increase the search cost temporarily.

1.1 Propositional satisfiability (SAT) complexity and hardness

It is proven that hard combinatorial SAT problems are the benchmarks that are used to test the efficiency, accuracy and optimality of a given algorithm [12]. As easy problems could be solved by any algorithm in a reasonable manner [21], which in turn does not reflect the real performance of a solving techniques. Therefore, studies since almost three decades focused on studying the hardness, complexity, and density of a countless number of SAT problems [6, 8, 15, 9]. Thus, a large distribution of hard problems was produced. These hard problems are categorized into two main divisions: 1) satisfiable and 2) un-satisfiable instances. Furthermore, in The International SAT solver competition (http"//www.satcompetition.org/) there are three sub divisions of the two main divisions: a) Industrial, b) Crafted, and c) Random instances. In another field of studies, researchers not only investigated whether a problem is satisfiable or not, they also studied how hard a problem is, as in [17, 21, 1].

1.2 Propositional satisfiability (SAT) dynamic solving techniques

Since the development of the Breakout heuristic [16], clause weighting dynamic local search (DLS) algorithms for SAT have been intensively investigated, and continually improved [5, 7]. However, the performance of these algorithms remained inferior to their non-weighting counterparts (e.g. [13]), until the more recent development of weight smoothing heuristics [24, 19, 11, 23]). Such algorithms now represent the state-of-the-art for stochastic local search (SLS) methods on SAT problems. Interestingly, the most successful DLS algorithms (i.e. DLM [24], SAPS [11], PAWS [23]), EWS [4], COVER [18] and recently CScore-SAT [3]) have converged on the same underlying weighting strategy: increasing weights on false clauses in a local minimum, then periodically reducing weights according to a problem specific parameter setting. Except for COVER which updates the edge weights in every step of the search.

However, a key issue with DLS algorithms is that their performance depend mainly on the efficiency of modifying the weights during the search, regardless of some other factors which may play a crucial role in their performance when applied for solving large and hard SAT problems such as Blocks World and Graph Coloring problems. For Instance, the size of backbones [22][1], the phase transition occurrence, and the density of a given problem.

Our study focuses on another factor and investigate its impact on the performance of DLS solving techniques. The question addressed in the current paper is that what happens to the weights when a clause and its neighboring clauses are satisfied?. For instance, if clause c_i is connected to *n* number of clauses (neighboring area of clause c_i, as discussed in sub-section 2.2) and by

assuming that clause c_i became satisfied, by flipping one of its literals l_{im}, where all its neighboring clauses are satisfied too, should clause c_i and its neighbors keep the weights?.

In the remainder of the paper we generally discuss the clause weighting most known algorithms such as SAPS, PAWS and DLM and DDFW. Then we discuss elaborately on DDFW technique as it is used for the purpose of the current study. Then, we show empirically the weights behaviors and movements during the search via an intensive experimentation on a broad range of benchmark SAT problems. Then we analyze the results and show the general outcome of the experiments. Finally, we conclude our work and some guidelines for future work are given.

2. CLAUSE WEIGHTING FOR SAT

Clause weighting local search algorithms for SAT follow the basic procedure of repeatedly flipping single literals that produce the greatest reduction in the sum of false clause weights. Typically, all literals are randomly initialized, and all clauses are given a fixed initial weight. The search then continues until no further cost reduction is possible, at which point the weight on all unsatisfied clauses is increased, and the search is resumed, punctuated with periodic weight reductions.

Existing clause weighting algorithms differ primarily in the schemes used to control the clause weights, and in the definition of the points where weight should be adjusted. Multiplicative methods, such as SAPS, generally adjust weights when no further improving moves are available in the local neighborhood. This can be when all possible flips lead to a worse cost, or when no flip will improve cost, but some flips will lead to equal cost solutions. As multiplicative real-valued weights have much finer granularity, the presence of equal cost flips is much more unlikely than for an additive approach (such as DLM or PAWS), where weight is adjusted in integer units. This means that additive approaches frequently have the choice between adjusting weight when no improving move is available, or taking an equal cost (flat) move.

Despite these differences, the three most well-known clause weighting algorithms (DLM [24], SAPS [11] and PAWS [23]) share a similar structure in the way that weights are updated:2 Firstly, a point is reached where no further improvement in cost appears likely. The precise definition of this point depends on the algorithm, with DLM expending the greatest effort in searching plateau areas of equal cost moves, and SAPS expending the least by only accepting cost improving moves. Then all three methods converge on increasing weights on the currently false clauses (DLM and PAWS by adding one to each clause and SAPS by multiplying the clause weight by a problem specific parameter $\alpha > 1$). Each method continues this cycle of searching and increasing weight, until, after a certain number of weight increases, clause weights are reduced (DLM and PAWS by subtracting one from all clauses with weight > 1 and SAPS by multiplying all clause weights by a problem specific parameter $\rho < 1$). SAPS is further distinguished by reducing weights probabilistically (according to a parameter P_{smooth}), whereas DLM and PAWS reduce weights after a fixed number of increases (again controlled by parameter). PAWS is mainly distinguished from DLM in being less likely to take equal cost or flat moves. DLM will take up to θ_1 consecutive flat moves, unless all available flat moves have

[1]back in 2001, Slaney et. al. [22] studied the impact of backbones in optimization and approximation problems. He concluded that in some optimization problems, backbones are correlated with the problem hardness. He also suggested that heuristic methods when used to identify backbones may reduce problem difficulty.

already been used in the last θ_2 moves. PAWS does away with these parameters, taking flat moves with a fixed probability of 15%, otherwise it will increase weight.

2.1 Divide and Distribute Fixed Weights

DDFW introduces two ideas into the area of clause weighting algorithms for SAT. Firstly, it evenly distributes a fixed quantity of weight across all clauses at the start of the search, and then escapes local minima by transferring weight from satisfied to unsatisfied clauses. The other existing state-of-the-art clause weighting algorithms have all divided the weighting process into two distinct steps: i) increasing weights on false clauses in local minima and ii) decreasing or normalizing weights on all clauses after a series of increases, so that weight growth does not spiral out of control. DDFW combines this process into a single step of weight transfer, thereby dispensing with the need to decide when to reduce or normalize weight. In this respect, DDFW is similar to the predecessors of SAPS (SDF [19] and ESG [20]), which both adjust and normalize the weight distribution in each local minimum. Because these methods adjust weight across all clauses, they are considerably less efficient than SAPS, which normalizes weight after visiting a series of local minima.3DDFW escapes the inefficiencies of SDF and ESG by only transferring weights between pairs of clauses, rather than normalizing weight on all clauses. This transfer involves selecting a single satisfied clause for each currently unsatisfied clause in a local minimum, reducing the weight on the satisfied clause by an integer amount and adding that amount to the weight on the unsatisfied clause. Hence DDFW retains the additive (integer) weighting approach of DLM and PAWS, and combines this with an efficient method of weight redistribution, i.e. one that keeps all weight reasonably normalized without repeatedly adjusting weights on all clauses.

Algorithm 1 DDFW($\mathcal{F}$, W_{init})

```
1: randomly instantiate each literal in F;
2: set the weight w_i for each clause c_i ∈ F to W_init;
3: while solution is not found and not timeout do
4:     find and return a list L of literals causing the greatest reduction in weighted cost Δw when
       flipped;
5:     if (Δw < 0) or (Δw = 0 and probability ≤ 15%) then
6:         randomly flip a literal in L;
7:     else
8:         for each false clause c_f do
9:             select a satisfied same sign neighbouring clause c_k with maximum weight w_k;
10:            if w_k < Init then
11:                randomly select a clause c_k with weight w_k ≥ W_init;
12:            end if
13:            if w_k > W_init then
14:                transfer a weight of two from c_k to c_f;
15:            else
16:                transfer a weight of one from c_k to c_f;
17:            end if
18:        end for
19:    end if
20: end while
```

[2]Additionally, a fourth clause weighting algorithm, GLSSAT [14], uses a similar weight update scheme, additively increasing weights on the least weighted unsatisfied clauses and multiplicatively reducing weights whenever the weight on any one clause exceeds a predefined threshold.

[3]Increasing weight on false clauses in a local minimum is efficient because only a small proportion of the total clauses will be false at any one time.

DDFW's weight transfer approach also bears similarities to the operations research sub-gradient optimization techniques discussed in [20]. In these approaches, Lagrangian multipliers, analogous to the clause weights used in SAT, are associated with problem constraints, and are adjusted in local minima so that multipliers on unsatisfied constraints are increased and multipliers on satisfied constraints are reduced. This symmetrical treatment of satisfied and unsatisfied constraints is mirrored in DDFW, but not in the other SAT clause weighting approaches (which increase weights and then adjust). However, DDFW differs from sub-gradient optimization in that weight is only transferred between pairs of clauses and not across the board, meaning less computation is required.

2.2 Exploiting Neighborhood Structure

Second and more original idea developed in DDFW, is the exploitation of neighborhood relationships between clauses when deciding which pairs of clauses will exchange weight.

We term clause c_i to be a neighbor of clause c_j, if there exists at least one literal $l_{im} \in c_i$ and a second literal $l_{jn} \in c_j$ such that $l_{im} = l_{jn}$ as in Fig 1. Furthermore, we term c_i to be a same sign neighbor of c_j if the sign of any $l_{im} \in c_i$ is equal to the sign of any $l_{jn} \in c_j$ where $l_{im} = l_{jn}$. From this it follows that each literal $l_{im} \in c_i$ will have a set of same sign neighboring clauses $C_{l_{im}}$. Now, if c_i is false, this implies all literals $l_{im} \in c_i$ evaluate to false. Hence flipping any l_{im} will cause it to become true in c_i, and also to become true in all the same sign neighboring clauses of l_{im}, i.e. it will increase the number of true literals, thereby increasing the overall level of satisfaction for those clauses. Conversely, l_{im} has a corresponding set of opposite sign clauses that would be damaged when l_{im} is flipped.

The reasoning behind the DDFW neighborhood weighting heuristic proceeds as follows: if a clause c_i is false in a local minimum, it needs extra weight in order to encourage the search to satisfy it. If we are to pick a neighboring clause c_j that will donate weight to c_i, we should pick the clause that is most able to pay. Hence, the clause should firstly already be satisfied. Secondly, it should be a same sign neighbor of c_i, as when c_i is eventually satisfied by flipping l_{im}, this will also raise the level of satisfaction of l_{im}'s same sign neighbors. However, taking weight from c_j only increases the chance that c_j will be helped when c_i is satisfied, i.e. not all literals in c_i are necessarily shared as same sign literals in c_j, and a non-shared literal may be subsequently flipped to satisfy c_i. The third criteria is that the donating clause should also have the largest store of weight within the set of satisfied same sign neighbors of c_i.

The intuition behind the DDFW heuristic is that clauses that share same sign literals should form alliances, because a flip that benefits one of these clauses will always benefit some other member(s) of the group. Hence, clauses that are connected in this way will form groups that tend towards keeping each other satisfied. However, these groups are not closed, as each clause will have clauses within its own group that are connected by other literals to other groups. Weight is therefore able to move between groups as necessary, rather than being uniformly smoothed (as in existing methods).

3. EXPERIMENTAL EVALUATION AND ANALYSIS

Problem	CNF			clausSize		# min size	# max size
	Atoms	Clauses	literals	min	max		
bw-large.d	6325	131973	294118	2	20	102580	25
logistics.c	1141	10719	27978	2	14	5846	46
ais10	181	3151	7255	2	10	2322	10
g125.17	2125	66272	134419	2	17	66147	125
g125.18	2250	70163	142326	2	18	70038	125
bmc-ibm-5	9396	41207	103560	1	53	73	1
bmc-ibm-10	59056	323700	853193	1	32	447	1
bcsp5080-h	750	26510	53670	2	15	26460	50
bcsp5080-m	750	26510	53670	2	15	26460	50
bcsp5040-h	750	26456	53562	2	15	26406	50
bcsp5040-m	750	26510	53670	2	15	26460	50
bw-large.b	1087	13772	31767	2	12	9735	11
f400-h	400	1720	5160	3	3	1720	1720
bcsp3040-h	300	5598	11436	2	10	5568	30
uf250-h	250	1065	3195	3	3	1065	1065
bmc-ibm-1	9685	55855	149765	1	39	49	1
uf400-h	400	1700	5100	3	3	1700	1700
uf100-m	100	430	1290	3	3	430	430
rocket	433	2902	5839	1	5	42	14
par32-3	3176	10297	2758	1	3	141	7128
par16-m	334	1332	3874	2	3	122	1210
hanoi6.shuff	4968	39666	98346	1	9	1020	126
flat200-h	600	2237	4674	2	3	2037	200
flat100-h	300	1117	2334	2	3	1017	100
flat100-m	300	1117	2334	2	3	1017	100
f1600-h	1600	6880	20640	3	3	6880	6880
f1600-m	1600	6873	20622	2	3	4	6869
f800-h	800	3440	10320	3	3	3440	3440
f800-m	800	3440	10320	3	3	3440	3440
g250.15	3750	233965	471180	2	15	233715	250
g250.39	7250	454622	915994	2	29	454372	250
huge	459	7054	15567	2	10	5682	5
bw_large.c	3016	50457	114314	2	16	37493	17

Table 1. structural information about some of the the original DIMACS 2005 problem sets. the problem were carefully selected taking to consideration their size, density, and connectivity.

As we stressed in section 2.1 and section 2.2, DDFW can exploit the neighboring structure of a given clause c_i and identify the weight alliances of c_i. These weight alliances act as the source of weights donors when a clause within the alliance need more weights. Also, they stabilize the process of weight transfer as they can lead to keeping weights within each allies as long as no weight transfer is needed, thus all the clauses within the neighborhood are satisfied. However, this has further implications as it could lead the search to get into one of the following scenarios that we discovered while investigating the weight transfer process: i) weights within a neighborhood (Allie) could not be transferred to another sub area of the search space. ii) weights of a specific neighborhood may keep circulating within their neighborhood. iii) if the level of connectivity between any two neighboring allies is low, weight transfer may suffer from stagnation, hence it can make the search process longer than it suppose to be.

In order to show the above three scenarios and their impact on the search process of any given DLM technique, we first studied the general structure of some benchmark problems. Table 1 illustrates the structure of the benchmark problems that were carefully selected based on their

size, complexity and hardness. We attempted to reproduce a problem set similar to that used in the random category of the SAT competition (as this is the domain where local search techniques have dominated). To do this we selected the 50 satisfiable k3 problems from the SAT2004 competition benchmark. Secondly, we obtained the 10 SATLIB quasi-group existence problems used in [2]. These problems are relevant because they exhibit a balance between randomness and structure. Finally, we obtained the structured problem set used to originally evaluate SAPS [11]. These problems have been widely used to evaluate clause weighting algorithms (e.g. in [23]) and contain a representative cross-section taken from the DIMACS and SATLIB libraries. In this set we also included 4 of the well-known DIMACS 16-bit parity learning problems.

For each selected problem we firstly show the number of atoms (variables) of the problem, the number of the clauses of the problem and the number of literals. Secondly, we show the minimum and the maximum number of literals that form a clause within the problem structure. Finally, we show the number of minimum sized clauses as will as the number of maximum sized clauses. We designed our experiment to be as follows:

- for each problem we ran DDFW 1000 run. Each run time out was set to 10,000,000 flips.
- in each run, we recorded the change of weights in every 10,000 flips.
- in every local minimum, we recorded whether DDFW heuristic selected a neighboring satisfied clause from the weight allie of the false clause, or it picked a satisfied clause, to be weight donor, randomly from outside the neighboring area of the false clause.
- plots were made for each problem to illustrates the change of weights of the false clauses from the starting point of the search process until the solution is found or it reaches a time out. This is done for all the figures included in the paper.

Table 2 show the detailed results of the runs. For each problem we firstly show the success rate (which reflect the percentage of whether a solution is found or not). Then we show the total number of local minima that DDFW heuristics found before reaching a global optima. Then we show the number of times the DDFW heuristic randomly picked a clause as a weight donor. Next we show the number of times that DDFW heuristic picked a neighboring satisfied clause as a weight donor from the false clause allies. Finally we show the the average number of times DDFW heuristic deterministically picked a clause as a weight donor.

In order to show the weights transfer and movements during the search we plotted the false clauses and their weights changes and the number of neighboring clauses of each false clause. This to show the relationship between the change of weights and the number of neighboring clauses of a false clause. Out of all problem sets we discuss four problems as they are of great importance to this work, namely, the Uniform Random 3SAT, the Parity problem, the Blocks World and the Graph Coloring problem because they explicitly show the previously mentioned three scenarios. These four problem sets are of different sizes and level of hardness. The Uniform Random 3SAT (uf100 and uf250) is the smallest set that has 100 variables and 430 clauses, where the Uniform Random 3SAT 250 has 250 variables and 1065 clauses. Fig 2 show the results for both problems. We can see that both problems were easy to solve. Also, the weight movement was smooth as their was enough neighboring clauses to donate weights to a false clause and more importantly when a sub area become satisfied, the connectivity between clauses allow the transfer

of weights easily (no weight stagnations occur). This is also true with the second problem, The Parity problem even that the level of hardness of the Parity 16 and the Parity 32 is higher than the Uniform Random 3SAT, as in Fig 3. What was experimentally interesting is the Graph Coloring problem (both the g125.17 and the g125.18 problems as in Fig 4, where g125.17 has 2125 variables 66272 clauses and the g125.18 has 2250 variables and 70163 clauses) where firstly, DDFW could not reach 100% success rate on the g125.17. Secondly, the figure show a clear gap between the movement of the weights. Which means the occurrence of weights stagnation. Thus, the connectivity between the clauses is either very low which make transferring weights among the clauses is limited to the clauses that are directly connected, or the connectivity of the clauses is very high which means a larger number of neighbors that could keep the weights for longer time and prevent other false clauses some where else in the search space from using them. Finally the Blocks World, Fig 5, which has 6325 variables and 131973 clauses., which has a similar weight transfer behaviors as the parity problem and the uniform random 3SAT problem with the exception of the level of hardness as the block world problem was harder to solve and the weights were moving more often during the search space.

problem	sRate	nLocalMinima	nRandomDist	nDeterminDist	avgDetr
bw-large.d	100	31441.5	4634.3	410747.3	86.9
logistics.c	100	232398.7	5276.9	445695.7	84.2
ais10	100	96772.7	2791.7	237724.4	70.4
g125.17	47	23438576	10545491.5	333747500.3	35.9
g125.18	100	81307.3	9347.9	508783.1	27.7
bmc-ibm-5	32	803981.7	812007.5	13719359	16.3
bmc-ibm-10	30	70030.7	439221	14571277.9	32.7
bcsp5080-h	90	12966448.1	3444963.8	107933199	39.4
bcsp5080-m	90	14887277.3	3920599.3	123967018.9	38.7
bcsp5040-h	23	20892491.8	5119784.4	241167931.3	46.6
bcsp5040-m	25	21307522.8	5296308.5	232836213.3	43.5
bw-large.b	100	4409.5	454.8	31528.1	67.0
f400-h	100	329514.4	16071.4	1583429	97.9
bcsp3040-h	100	3565452.6	478237.2	20765453.3	44.1
uf250-h	100	248885.2	7706.3	757378.6	97.6
bmc-ibm-1	70	1100209.7	439064.5	17506230.3	39.5
uf400-h	100	99885.8	5420.5	527031.4	97.3
uf100-m	100	664.6	18.5	1570.9	85.5
rocket	68	2027948.8	2302130.7	76669187.7	33
par32-3	89	793067.8	299015.4	11539640.6	38.1
par16-m	100	805116.6	72261.5	7064764.5	97.3
hanoi6.shuff	54	1067162.8	166107.2	15080491.9	90.2
flat200-h	100	756983.1	1635948.9	4709022	2.1
flat100-h	100	6779.6	7037.9	23926.1	2.4
flat100-m	100	38778	45406.8	152577.6	3.2
f1600-h	100	811342.5	126567.6	12453381.6	97.9
f1600-m	100	139874.9	19431.5	1900084.9	97.9
f800-h	100	447944.9	38308.9	3748470.3	97.4
f800-m	100	191996.7	18452	1820062.8	98.2
g250.15	100	20.5	734.6	186.4	0
g250.29	0	13676053.1	6477528.2	306931973	47.6
huge	100	986.6	118.6	6070.7	55.3
bw_large.c	100	53580.9	5654.3	490703	83.8

Table 2. The table show the number of successful tries made by DDFW, the number of local minima faced the search, the number of random weights distribution during the search, the number of deterministic weight distribution and the average number of determinstc weight distribution. Weights behaviors and movements during the search.

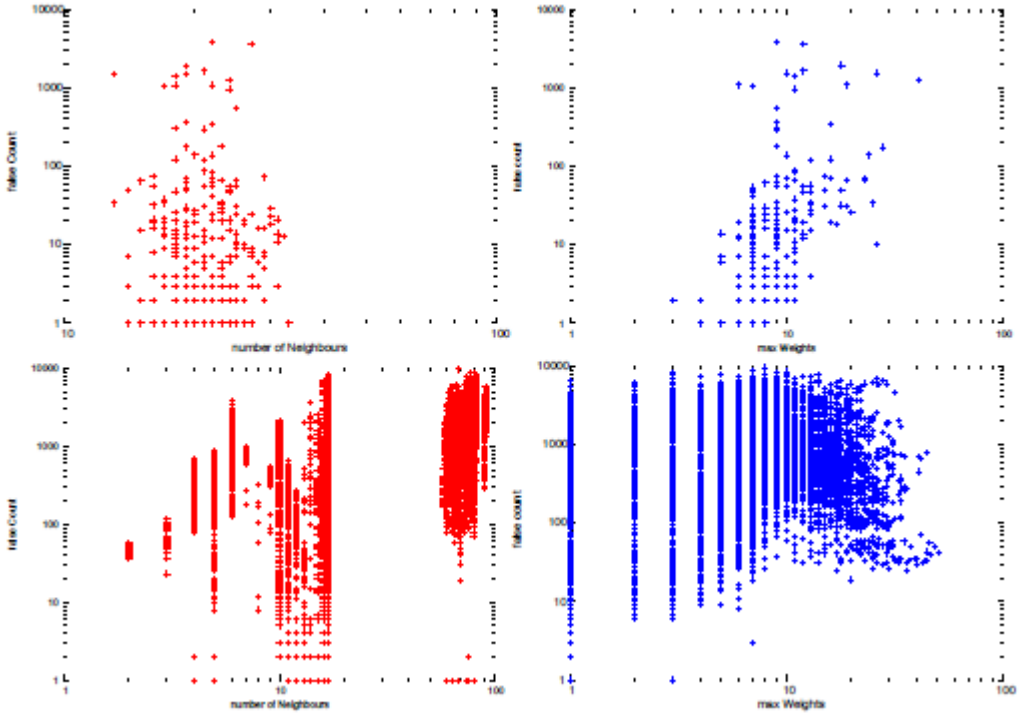


Fig. 2. false clauses and their weights during the search, the uf100 problem top and the uf250 bottom

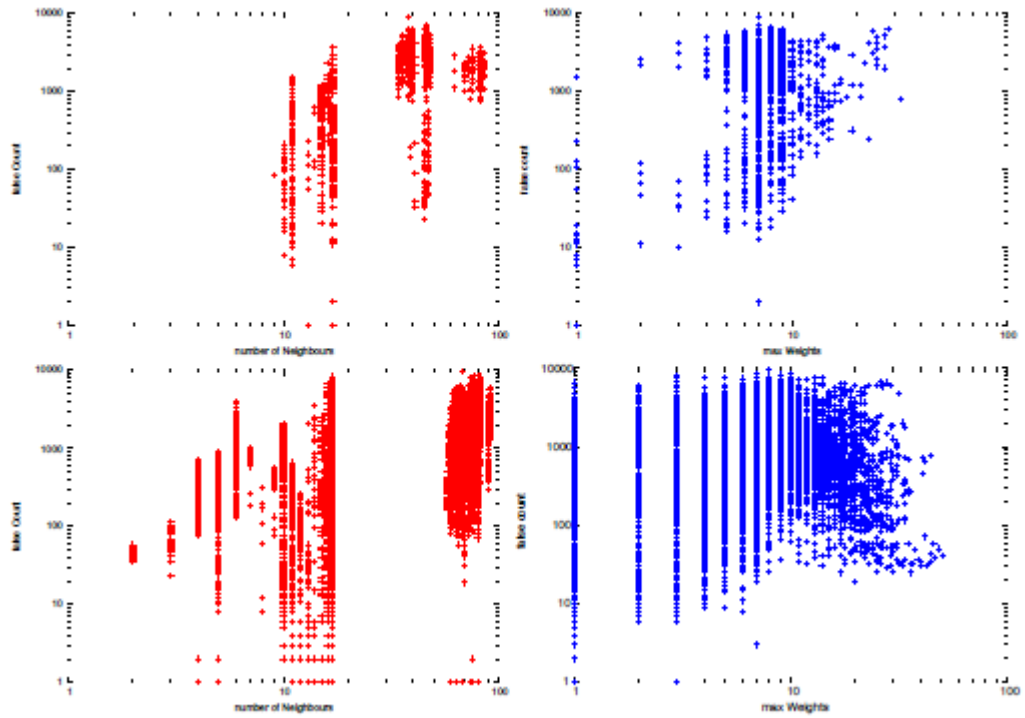


Fig. 3. false clauses and their weights during the search, the par16 top and par32 bottom

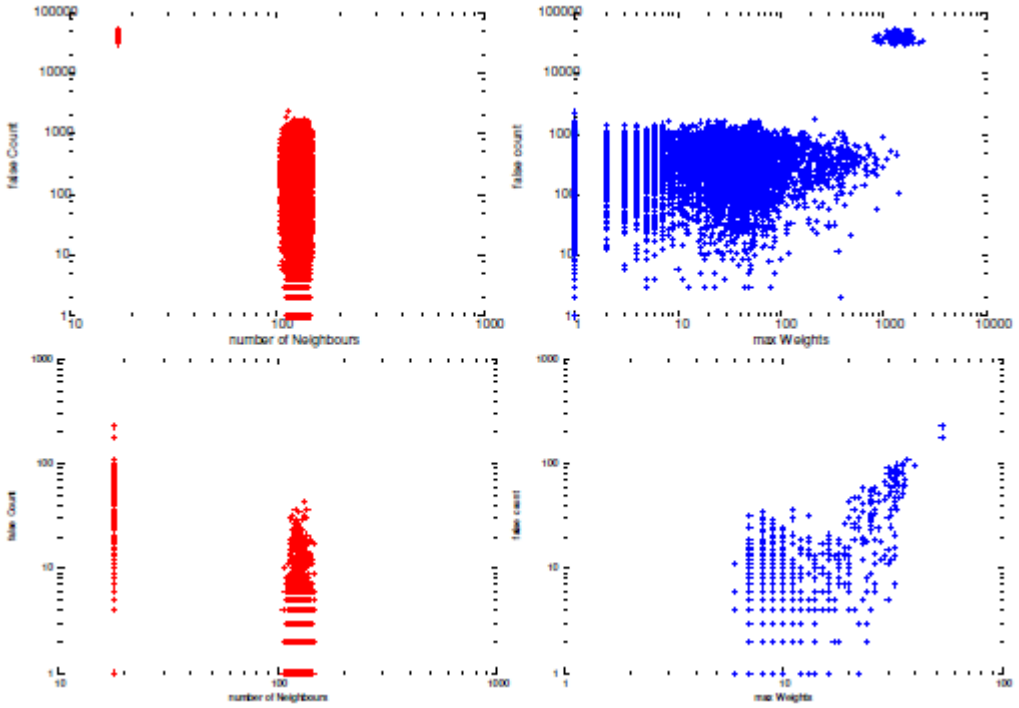


Fig. 4. false clauses and their weights during the search, the g125-17 top and the g125-18 bottom

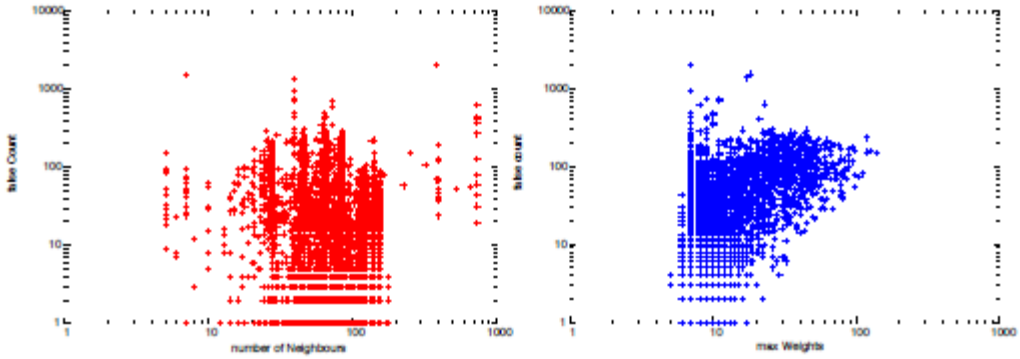


Fig. 5. false clauses and their weights during the search, the bw-d.large

4. CONCLUSION

As a conclusion, DLS weighting techniques performance could suffer from weight stagnation that leads to the slowness of a chosen techniques. Our experiments show that these weight stagnations are not general to all problems but rather they are a problem specific characteristic. We have looked into each problem characteristic such as its hardness, complexity, and density. As a result, each problem characteristics may contribute to the occurrence of weight stagnations in some stages during the search. The DDFW algorithm is a relatively simple application of neighborhood weighting, and further experiments (not reported here) indicate more complex heuristics can be more effective on individual problems. In particular, we have looked at adjusting the amount of weight that is redistributed and allowing DDFW to randomly pick donor clauses according to a noise parameter. However, we have yet to discover a general neighborhood heuristic as effective

as DDFW over the range of problems considered. In future work we consider it will be promising to extend a DDFW-like approach to handle weight stagnations via adjusting weights in stagnated alliances regardless of whether they are satisfied or not. This could be done by improving the exploitation of neighboring areas.

ACKNOWLEDGEMENT

The authors would like to acknowledge the financial support of the Scientific Research Committee at Petra University. Also we would like to thank all faculty members of the Information Technology faculty who contributed directly and indirectly to this work.

REFERENCES

[1] D. Achlioptas, C. Gomes, H. A. Kautz, and B. Selman, \Generating satisfiable instances," in Proceedings of 17th AAAI, 2000, pp. 256{261.

[2] Anbulagan, D. Pham, J. Slaney, and A. Sattar, \Old resolution meets modern SLS," in Proceedings of 20th AAAI, 2005, pp. 354{359.

[3] S. Cai, C. Luo, and K. Su, \Scoring functions based on second level score for k-sat with long clauses," J. Artif. Intell. Res. (JAIR), vol. 51, pp. 413{441, 2014.

[4] S. Cai, K. Su, and Q. Chen, \EWLS: A new local search for minimum vertex cover," in Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010, 2010.

[5] B. Cha and K. Iwama, \Adding new clauses for faster local search," in Proceedings of 13th AAAI, 1996, pp. 332{337.

[6] S. A. Cook, \The complexity of theorem-proving procedures," in Proceedings of the Third Annual ACM Symposium on Theory of Computing, ser. STOC'71. New York, NY, USA: ACM, 1971, pp. 151{158. [Online]. Available: http://doi.acm.org/10.1145/800157.805047

[7] J. Frank, \Learning short-term clause weights for GSAT," in Proceedings of 15th IJCAI, 1997, pp. 384{389.

[8] M. R. Garey and D. S. Johnson, Computers and Intractability; A Guide to the Theory of NP-Completeness. New York, NY, USA: W. H. Freeman & Co., 1990.

[9] I. P. Gent and T. Walsh, \The hardest random SAT problems," in KI-94: Advances in Artificial Intelligence, 18th Annual German Conference on Artificial Intelligence, Saarbrucken, Germany, September 18-23, 1994, Proceedings, 1994, pp. 355{366.

[10] H. Hoos and T. Stulze, Stochastic Local Search. Cambridge, Massachusetts: Morgan Kaufmann, 2005.

[11] F. Hutter, D. Tompkins, and H. Hoos, \Scaling and Probabilistic Smoothing: Efficient dynamic local search for SAT," in Proceedings of 8th CP, 2002, pp. 233{248.

[12] M. Jarvisalo, D. Le Berre, O. Roussel, and L. Simon, \The international SAT solver competitions," AI Magazine, vol. 33, no. 1, pp. 89{92, 2012.

[13] D. McAllester, B. Selman, and H. Kautz, \Evidence for invariants in local search," in Proceedings of 14th AAAI, 1997, pp. 321{326.

[14] P. Mills and E. Tsang, \Guided local search applied to the satisfiability (SAT) problem," in Proceedings of 15th ASOR, 1999, pp. 872{883.

[15] D. Mitchell, B. Selman, and H. Levesque, \Hard and easy distributions of sat problems," in ghiuyguyigy, 1992, pp. 459{465.

[16] P. Morris, \The Breakout method for escaping from local minima," in Proceedings of 11th AAAI, 1993, pp. 40{45.

[17] P. Prosser, \Binary constraint satisfaction problems: Some are harder than others," in ECAI. PITMAN, 1994, pp. 95{95.

[18] S. Richter, M. Helmert, and C. Gretton, \A stochastic local search approach to vertex cover," in KI 2007: Advances in Artificial Intelligence, 30th Annual German Conference on AI, KI 2007, Osnabruck, Germany, September 10-13, 2007, Proceedings, 2007, pp. 412{426.

[19] D. Schuurmans and F. Southey, \Local search characteristics of incomplete SAT procedures," in Proceedings of 10th AAAI, 2000, pp. 297{302.

[20] D. Schuurmans, F. Southey, and R. Holte, \The exponentiated subgradient algorithm for heuristic boolean programming," in Proceedings of 17th IJCAI, 2001, pp. 334{341.

[21] B. Selman, D. Mitchell, and H. Levesque, \Generating hard satisfiability problems," Artificial Intelligence, vol. 81, pp. 17{29, 1996.

[22] J. Slaney and T. Walsh, \Backbones in optimization and approximation," in Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume1, ser. IJCAI'01, 2001, pp. 254{259.

[23] J. Thornton, D. N. Pham, S. Bain, and V. Ferreira Jr., \Additive versus multiplicative clause weighting for SAT," in Proceedings of 19th AAAI, 2004, pp. 191{196.

[24] Z. Wu and B. Wah, \An efficient global-search strategy in discrete Lagrangian methods for solving hard satisfiability problems," in Proceedings of 17th AAAI, 2000, pp. 310{315

CORROSION DETECTION USING A.I. : A COMPARISON OF STANDARD COMPUTER VISION TECHNIQUES AND DEEP LEARNING MODEL

Luca Petricca[*1], Tomas Moss[2], Gonzalo Figueroa[2] and Stian Broen[1]

[1]Broentech Solutions A.S., Horten, Norway
*lucap@broentech.no
[2]Orbiton A.S. Horten, Norway,
info@orbiton.no

ABSTRACT

In this paper we present a comparison between standard computer vision techniques and Deep Learning approach for automatic metal corrosion (rust) detection. For the classic approach, a classification based on the number of pixels containing specific red components has been utilized. The code written in Python used OpenCV libraries to compute and categorize the images. For the Deep Learning approach, we chose Caffe, a powerful framework developed at "Berkeley Vision and Learning Center" (BVLC). The test has been performed by classifying images and calculating the total accuracy for the two different approaches.

KEYWORDS

Deep Learning; Artificial Intelligence; Computer Vision; Caffe Framework; Rust Detection.

1. INTRODUCTION

Bridge inspection is one important operation that must be performed periodically by public road administrations or similar entities. Inspections are often carried out manually, sometimes in hazardous conditions. Furthermore, such a process may be very expensive and time consuming. Researchers [1, 2] have put a lot of effort into trying to optimize such costly processes by using robots capable of carrying out automatic bridge maintenance, reducing the need for human operators. However such a solution is still very expensive to develop and carry out. Recently companies such as Orbiton AS have started providing bridge inspection services using drones (multicopters) with high resolution cameras. These are able to perform and inspect bridges in many adverse conditions, such as with a bridge collapse[3], and/or inspection of the underside of elevated bridges. The videos and images acquired with this method are first stored and then subsequently reviewed manually by bridge administration engineers, who decide which actions are needed. Even though this sort of automation provides clear advantages, it is still very time consuming, since a physical person must sit and watch hours and hours of acquired video and images. Moreover, the problem with this approach is twofold. Not only are man-hours an issue

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 91–99, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60608

for infrastructure asset managers, so is human subjectivity. Infrastructure operators are nowadays requesting methods to analyse pixel-based datasets without the need for human intervention and interpretation. The end result desired is to objectively conclude if their assets present a fault or not. Currently, this conclusion varies according to the person doing the image interpretation and analysis. Results are therefore inconsistent, since the existence of a fault or not is interpreted differently depending on the individual. Where one individual sees a fault, another may not. Developing an objective fault recognition system would add value to existing datasets by providing a reliable baseline for infrastructure asset managers. One of the key indicators most asset managers look for during inspections is the presence of corrosion. Therefore, this feasibility study has focused on automatic rust detection. This project created an autonomous classifier that enabled detection of rust present in pictures or frames. The challenge associated with this approach was the fact that the rust has no defined shape and colour. Also, the changing landscape and the presence of misleading object (red coloured leaves, houses, road signs, etc) may lead to miss-classification of the images. Furthermore, the classification process should still be relatively fast in order to be able to process large amount of videos in a reasonable time.

2. APPROACH USED

Some authors tried to solve similar problems using "watershed segmentation" [4] for coated materials, supervised classification schemes [5-6] for cracks and corrosion in sewer pipes and metal, or Artificial Neural Networks [7] for corrosion in vessels. We decided to implement one version of classic computer vision (based on red component) and one deep learning model and perform a comparison test between the two different approaches. Many different frameworks and libraries are available for both the classic computer vision techniques and the Deep Learning approach.

2.1. Classic Computer Vision Technique

For almost two decades, developers in computer vision have relied on OpenCV[8] libraries to develop their solutions. With a user community of more than 47 thousand people and estimated number of downloads exceeding 7 million, this set of >2500 algorithms [8] and useful functions can be considered standard libraries for image and video applications. The library has two interfaces, C++ and Python. However, since Python-OpenCV is just a wrapper around C++ functions (which perform the real computation intensive code), the loss in performance by using Python interface is often negligible. For these reasons we chose to develop our first classifier using this set of tools. The classifier was relatively basic. Since a corroded area (rust) has no clear shape, we decided to focus on the colours, and in particular the red component. After basic filtering, we changed the image colour space from RGB to HSV, in order to reduce the impact of illumination on the images[6]. After the conversion we extracted the red component from the HSV image (in OpenCV, Hue range is [0,179], Saturation range is [0,255] and Value range is [0,255]). Since the red components is spread in a non-contiguous interval (range of red color in HSV is around 160-180 and 0-20 for the H component) we had to split the image into two masks, filter it and then re-add them together. Moreover, because not all the red interval was useful for the rust detection, we tried to empirically narrow down the component in order to find the best interval that was not result in too many false positives. After extensive testing we found the best interval in rust detection, to be 0-11 and 175-180. Also we flattened the S and I component to the range 50-255. This mask was then converted into black and white and the white pixels were counted. Every image having more than 0.3% of white pixels was finally classified as

"rust", while having less than 0.3% of white pixels indicated a "non-rust" detection. Below are some snippets of the classification code:

```
# define range of red color in HSV 160-180 and 0-20
lower_red = np.array([0,50,50])
upper_red = np.array([11,255,255])
lower_red2 = np.array([175,50,50])
upper_red2 = np.array([179,255,255])

# Threshold the HSV image to get only red colors

mask1 = cv2.inRange(hsv, lower_red, upper_red)
mask2 = cv2.inRange(hsv, lower_red2, upper_red2)
mask=mask1+mask2
ret,maskbin = cv2.threshold(mask,
      127,255,cv2.THRESH_BINARY)

#calculate the percentage
height, width = maskbin.shape
size=height * width
percentage=cv2.countNonZero(maskbin)/float(size)
if percentage>0.003:
    return True
else:
    return False
```

2.2. Deep Learning Model

The second approach was based on artificial intelligence, in particular using Deep Learning methods. This approach is not new. The mathematical model of back-propagation was first developed in '70s and was originally reused by Yann LeCun in [9]. This was one of the first real applications of Deep Learning. However a major step forward was made in 2012 when Geoff Hinton won the imageNet competition by using Deep Learning network, outperforming other more classic algorithms. Among many frameworks available such as torch [10], theano library for python, or the most recent tensorflow [11] released by google, we chose caffe from "*Berkeley Vision and Learning Center"* (BVLC)[12]. This framework is specifically suited for image processing, offering good speed and great flexibility. It also offers the opportunity to easily use clusters of GPUs support for model training which could be useful in the case of large networks. Furthermore, it is released under a BSD 2 license.The first step was to collect a good dataset to be used to train the network. We were able to collect around 1300 images for the "rust" class and 2200 images for the "non-rust" class. Around 80% of the images were used for the training set, while the rest was used for the validation set. Since the dataset was relatively small, we decided to fine tune an existing model called "*bvlc_reference_caffenet*" which is based on the AlexNet model and released with license for unrestricted use. In fine tuning, the framework took an already trained network and adjusted it (resuming the training) using the new data as input. This technique provides several advantages. First of all, it allows the reuse of previously trained networks, saving a lot of time. Furthermore, since the "*bvlc_reference_caffenet*" has been already pre-trained with 1 Million images, the network has prior "knowledge" of the correct weight

parameters for the initial layers. We could thus reuse that information and avoid over-fitting problems (excessively complex model and not enough data to constrain it). The last layer of the model was also modified to reflect the rust requirements. In particular the layer 8 definition was changed to:

```
layer {
  name: "fc8_orbiton"
  type: "InnerProduct"
  bottom: "fc7"
  top: "fc8_orbiton"
  param {
    lr_mult: 10
    decay_mult: 1
  }
  param {
    lr_mult: 20
    decay_mult: 0
  }
  inner_product_param {
    num_output: 2
    weight_filler {
      type: "gaussian"
      std: 0.01
    }
```

Notice that the learning rate (lr) multiplier was set to 10 in order to make the last layer weight parameters "move" more in respect to the other layers where the learning rate multiplier was set to 1 (because they were already pre-trained). Also we set up the number of outputs to two to reflect our two categories "rust"/"non-rust". The images were resized to 256x256 pixels and the mean file for the new set of images was recalculated before we trained the model. The training process was performed with a learning rate of 0.00005 with 100.000 iterations in total, performed on an Ubuntu 14.04 machine with GPU CUDA support. The hardware included I7 skylake CPU and Nvidia GTX 980 Ti GPU. The training process took around 2 days.

3. TESTS

Test and comparison of the trained model was performed by writing a small classification script in Python and using it for classifying a new image set. This new set of images was different from the one used in the Deep Learning training and validation steps and consisted of 100 images, divided into two groups of 37 images of "rust" and 63 images of "non-rust". Images were chosen as a mix of real case image (picture of bridges, metal sheets, etc) and other added just to trick the algorithm such as images from desert landscape or images of red apple trees. In Figure 1 shows some example of the images used.

Figure 1: Example of test images

4. RESULTS

Results of the test were divided into two groups:

1. False Positive: The images of “non-rust” which were wrongly classified as “rust”
2. False Negative: The images of “rust” which were wrongly classified as “non-rust”

For each algorithm developed, we counted the number of false positives and false negatives occurrences. The partial accuracy for the classification in each class was also calculated based on the total images of that class. For example OpenCV had 4 false negative images on 37 “rust” image. This implies that 33 images over 37 were correctly classified as “rust” giving a total accuracy of for the “rust class” of:

$$\frac{37-4}{37} x\,100 = 89.1\%$$

Similarly, for the “non-rust” class the partial accuracy is given by the correctly classified images of “non-rust” (36) over the total “non-rust” images (63), giving an partial accuracy for the “non-rust” class of 57%. We also included a total accuracy for the total number of correctly classified images over the total. In this case OpenCV classified correctly 69 images out of 100 (69%).
We repeated the same calculation for the Deep Learning model and the results are reported in the column two of Table 1.

The Deep Learning classifier also provides a probability associated with each prediction. This number reflects how confident the model is that the prediction is correct. Among the 100 images, 15 of those had a probability below 80%. We discarded those images and recalculated the accuracy values (third column). This also means that for the 15% of the image, the Deep Learning model was “undecided” on how to classify it. A complete summary of the results is reported in Table 1.

Table 1. Models comparison: resume table

	OpenCV	**Deep Learning**	**Deep Learning Probability >0.8**
False Positive	27/ 63	14/63	5/51
Partial Accuracy for "non-rust"	57%	78%	90%
Number False Negative	4/ 37	8/37	7/34
Partial Accuracy for "rust"	89%	78%	79.4%
Total Accuracy (correctly classified/total of images)	**69%**	**78%**	**88%**

5. DISCUSSION

The results show a few interesting facts about the two approaches. The OpenCV based model showed a total accuracy (in all the images) of 69%. According to our expectations, it presented a reduced accuracy (57%) for the "non-rust" classification, while it had great accuracy for the "rust" classification (almost 90%). The reason for this is pretty clear: all the "rusty" images had red components, so it was easy for the algorithm to detect it. However, for the "non-rust" class, the presence of red pixels does not necessary imply the presence of rust. So when we pass a red apple picture, the model just detected red component and misclassified it as "rust", reducing the "non-rust" accuracy. All the four pictures in Figure 1 for example, have been classified by the OpenCV algorithm as "rust", while only two of them are actually correct. The few false negatives involved (where there was rust but it was not correctly detected), seemed were due mainly to the bad illumination of the image, problems associated with colour (we also provided few out of focus test images), or the rust spot was too small (less than 0.3% of the image).

For the Deep Learning Algorithm, things get more interesting. Indeed, we noticed a more uniform accuracy (78% in total) between the "rust" detection and the "non-rust" detection (78% in both the cases). In this case the model is also more difficult to "trick": For example all the images in Figure 1 were correctly classified from the model, despite the fact that we never used any apple or desert image during the training process. So we analysed the most common pictures where it failed, to get some useful information from it. In Figure 2 are reported a few examples of "non rust" picture, wrongly classified as "rust" from the Deep Learning model. It is important to mention that all the pictures in Figure 2 were also misclassified by the OpenCV algorithm. We believe that in the first and last image, the presence of red leaves led the algorithm to believe that it was rust. In the second image, the rust on the concrete section was wrongly classified as "rust" in metal. The third image was more difficult to explain, however a reasonable explanation may be the presence of the mesh pattern in the metal and a little reddish drift of the colours.

In Figure 3 are shown some examples of pictures classified as "non-rust", while there was actually rust. It seems that they have something in common, so the reason for the misclassification may be that the system has "never seen" something similar. The two images on the top were correctly categorized from the OpenCV, while the two bottom ones were not.
A few considerations about the confidence level of the Deep Learning model are also interesting. We noticed that for most of the images the model gave us a "confident rate" above 80%. In 15% of the images, this confidence was less than 80%. If we analyse this 15% in detail, we discovered that actually 9 of those were wrongly classified, while only 6 were correct. By discarding these

images we were able to increase the total accuracy from 78%to 88%. So the model already provides us with a useful and reliable parameter that can be directly used to improve the overall accuracy.

Figure 2: Example of picture wrongly classified as Rust from the Deep Learning model

Figure 3: Examples of pictures wrongly classified as No-Rust from the Deep Learning model

Even more interesting are the results from a possible combination of the two algorithms. In 77 images both the algorithms agree on the result. Of these 77 images, only 12 (3+9) were wrong. This would have given us a partial accuracy of 92% for the “rust” and 78% for the “non-rust”.
Another interesting solution would be to use the OpenCV to filter out the “non-rust” image, and then pass the possible rust image to the Deep Learning model. In this case we could potentially

create a system much more accurate with an accuracy of 90% of "rust" and 81% for the "non-rust". More complex solutions are also possible, for example by discarding from the "possible rust", where the Deep Learning model has a confidence level less than 80%.

6. CONCLUSIONS

In this paper we presented a comparison between two different models for rust detection: one based on red component detection using OpenCV library, while the second one using Deep Learning models. We trained the model with more than 3500 images and tested with a new set of 100 images, finding out that the Deep Learning model performs better in a real case scenario. However for a real application, it may be beneficial to include both the systems, with the OpenCV model used just for removing the false positives before they are passed to the Deep Learning method. Also, the OpenCV based algorithm may also be useful for the classification of images where the Deep Learning algorithm has low confidence. In future work we will seek to refine the model and train it with a new and larger dataset of images, which we believe would improve the accuracy of the Deep Learning model. Subsequently, we will do some field testing using real time video from real bridge inspections.

ACKNOWLEDGEMENTS

We would like to thank Innovation Norway and Norwegian Centres of Expertise Micro- and Nanotechnology (NCE-MNT) for funding this project and Statens vegvesen and Aas-Jakobsen for providing image datasets. Some of the pictures used were also downloaded from pixabay.com

REFERENCES

[1] A.Leibbrandt et all. "Climbing robot for corrosion monitoring of reinforced concrete structures" DOI: 10.1109/CARPI.2012.6473365 2nd International Conference on Applied Robotics for the Power Industry (CARPI), 2012

[2] Jong Seh Lee, Inho Hwang, Don-Hee Choi Sang-Hyun Hong, "Advanced Robot System for Automated Bridge Inspection and Monitoring", IABSE Symposium Report 12/2008; DOI: 10.2749/222137809796205557.

[3] "Bridge blown up, to be built anew", newsinenglish.no, http://www.newsinenglish.no/2015/02/23/bridge-blown-up-to-be-built-anew/

[4] Gang Ji, Yehua Zhu, Yongzhi Zhang, "The Corroded Defect Rating System of Coating Material Based on Computer Vision" Transactions on Edutainment VIII Springer Volume 7220 pp 210-220

[5] F Bonnín-Pascual, A Ortiz, "Detection of Cracks and Corrosion for Automated Vessels Visual Inspection", A.I. Research and Development: Proceedings of the 13th conference.

[6] N. Hwang, H. Son, C. Kim, and C. Kim, "Rust Surface Area Determination Of Steel Bridge Component For Robotic Grit-Blast Machine", isarc2013Paper305.

[7] Moselhi, O. and Shehab-Eldeen, T. (2000). "Classification of Defects in Sewer Pipes Using Neural Networks." J. Infrastruct. Syst., 10.1061/(ASCE)1076-0342(2000)6:3(97), 97-104.

[8] Open CV, Computer Vision Libraries: OpenCV.org

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel: "Backpropagation Applied to Handwritten Zip Code Recognition, Neural Computation", 1(4):541-551, Winter 1989.

[10] Torch, Scientific Computing Framework http://torch.ch/

[11] Tensor Flow, an open source software library for numerical computation, https://www.tensorflow.org/

[12] Caffe Deep Learning Framework, Berkeley Vision and Learning Center (BVLC), http://caffe.berkeleyvision.org/.

INTENTIONAL BLANK

NONLINEAR EXTENSION OF ASYMMETRIC GARCH MODEL WITHIN NEURAL NETWORK FRAMEWORK

Josip Arnerić[1] and Tea Poklepović[2]

[1]Faculty of Economics and Business Zagreb,
Department of Statistics, Trg J. F. Kennedyja 6, 1000 Zagreb, Croatia
jarneric@efzg.hr
[2]Faculty of Economics Split, Department of Quantitative Methods,
Cvite Fiskovića 5, 21000 Split, Croatia
tpoklepo@efst.hr

ABSTRACT

The importance of volatility for all market participants has led to the development and application of various econometric models. The most popular models in modelling volatility are GARCH type models because they can account excess kurtosis and asymmetric effects of financial time series. Since standard GARCH(1,1) model usually indicate high persistence in the conditional variance, the empirical researches turned to GJR-GARCH model and reveal its superiority in fitting the asymmetric heteroscedasticity in the data. In order to capture both asymmetry and nonlinearity in data, the goal of this paper is to develop a parsimonious NN model as an extension to GJR-GARCH model and to determine if GJR-GARCH-NN outperforms the GJR-GARCH model.

KEYWORDS

conditional volatility, GARCH model, GJR model, Neural Networks, emerging markets

1. INTRODUCTION

Modelling volatility, i.e. returns fluctuations, has been a topic of interest to economic and financial researchers. Portfolio managers, option traders and market makers are all interested in volatility forecasting in order to get higher profits or less risky positions.

The most popular models in modelling volatility are generalized autoregressive conditional heteroskedasticity (GARCH) type models which can account excess kurtosis and asymmetric effects of high frequency data, time varying volatility and volatility clustering. The first autoregressive conditional heteroscedasticity model (ARCH) was proposed by Engle [1] who won a Nobel Prize in 2003 for his contribution to modelling volatility. The model was extended by Bollerslev [2] by its generalized version (GARCH). However, standard GARCH(1,1) model usually indicates high persistence in the conditional variance, which may originate from structural changes in the variance process. Hence the estimates of a GARCH model suffer from a

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 101–111, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60609

substantial upward bias in the persistence parameters. Also, it is often difficult to predict volatility using traditional GARCH models because the series is affected by different characteristics: non-stationary behaviour, high persistence in the conditional variance, asymmetric behaviour and nonlinearity. Due to practical limitations of these models different approaches have been proposed in the literature. Some of them are developed for resolving the asymmetric behaviour problem and some of them the nonlinearity in variance. Diebold [3] found that volatility models that fail to adequately incorporate nonlinearity are subject to an upward bias in the parameter estimates which results in strong forms of persistence that occurs especially in high volatility periods in financial time series and this influences the out-of-sample forecasts of single regime type GARCH models. The empirical researches reveal that among asymmetric models, Glosten, Jaganntahn and Runkle's [4] sign-ARCH model, i.e. GJR-GARCH model outperforms all the other GARCH-type models. Moreover, to account for nonlinearity, in recent researches much attention is given to neural network models (NN) in forecasting volatility.

The NNs are a valuable tool for modelling and prediction of time series in general ([5]; [6]; [7]; [8]; [9]; [10]; [11]; [12]; [13]). Most financial time series indicate existence of nonlinear dependence, i.e. current values of a time series are nonlinearly conditioned on information set consisting of all relevant information up to and including period $t-1$ ([14]; [15]; [16]; [17]). The feed-forward neural networks (FNN), i.e. multilayer perceptron, are most popular and commonly used. They are criticized in the literature for the high number of parameters to estimate and they are sensitive to overfitting ([18]; [19]).

The objective of this paper is to develop a parsimonious NN model as an extension to GJR-GARCH model which will capture the nonlinear relationship between past return innovations and conditional variance. The second objective of this paper is to determine if GJR-GARCH-NN model outperforms GJR-GARCH models when there is high persistence of the conditional variance. This paper contributes to existing literature in several ways. Firstly, this paper introduces NN as semiparametric approach, which combines flexibility of nonparametric methods and the interpretability of parameters of parametric methods, and attractive econometric tool for conditional volatility forecasting. NN models have continuously been observed as a nonparametric method relying on automatically chosen NN provided by various software tools, which is unjustified from the econometric perspective. Therefore, in this paper the "black box" will be opened. Secondly, in this paper new NN model is defined, as an extension to GJR-GARCH models, and estimated. Although this paper relies on paper from Donaldson and Kamstra [20] it contributes to the literature by estimating an additional parameter λ which was previously set in advance. Finally, this paper contributes to the literature of emerging market economies with the newest data.

The remainder of this paper is organized as follows. Section 2 presents the literature review. Section 3 describes the data and methodology. Section 4 presents the obtained empirical results and discussion. Finally, some conclusions and directions for future research are provided in Section 5.

2. LITERATURE REVIEW

Donaldson and Kamstra [20] in their paper construct a seminonparametric nonlinear GARCH model, based on NN approach, and evaluate its ability to forecast stock return volatility on stock exchanges in London, New York, Tokyo and Toronto using daily stock returns from 1970 to

1990. They compared this constructed NN model with performances of other most commonly used volatility models, i.e. GARCH, EGARCH and GJR-GARCH model, in in- and out-of-sample comparison and within different markets. The results reveal that GJR-GARCH model fits the asymmetric heteroscedasticity in the data better than GARCH and EGARCH models, however the best performing model of all seems to be the newly introduced NN model. The authors present the new methodology which is applied in advancing markets, however, the properties of selected methodology are not yet tested in emerging markets. Moreover, the number of hidden neurons is obtained by selecting the best alternative model in the grid [0,5] parameter space using Schwarz information criterion. In this paper the number of hidden units in three-layer NN is specified in advance for the more suitable comparison between models. This paper contributes in estimating an additional parameter λ which was in presented paper set in advance.

Teräsvirta et al [13] present similar methodology as [20] based on Medeiros et al [11] approach and use it as AR-NN type model showing the potential of their proposed modelling approach in two applications: sunspot series and US unemployment series. Moreover, they clearly combine the NN model so as to be able to explain it as econometric model.

Bildirici and Ersin [21] analyse the volatility of stock returns on Istanbul Stock Exchange (ISE) in period from 1987 to 2008 using daily closing prices of ISE 100 index. They compare and combine GARCH, EGARCH, GJR-GARCH, TGARCH, NGARCH, SAGARCH, PGARCH, APGARCH, NPGARCH with NN models in their forecasting abilities. The NN models are retrained with conjugate gradient descent algorithm after the training with backpropagation. They conclude that NN models improved the generalization and forecasting ability of GARCH models. In their paper the models are not properly explained from an econometric perspective, nor are the findings explained from the perspective to the real time data. Moreover, the parameters of the models are nor presented or explained.

Their later paper, Bildirici and Ersin [22], relies on paper from [20] and [21] to analyse the nonlinearity and leptokurtic distribution of stock returns on ISE in period from 1986 to 2010 and benefits from both LSTAR and NN type of nonlinearity, i.e. this paper proposes several LSTAR-GARCH-NN family models. GARCH, FI-GARCH, APGARCH and FIAPGARCH models are augmented with a NN model. They conclude that extended GARCH models forecast better than GARCH models; LSTAR-LST-GARCH show significant improvement in out-of-sample forecasting; MLP-GARCH models provide similar results to LSTAR-LST-GARCH models; LSTAR-LST-APGARCH-MLP model provided the best overall performance. To estimate NN models, the number of hidden neurons ranges from 3 to 10 and the best model is selected based on MSE or RMSE. Moreover, each of the selected model architecture is estimated 20 times for 8 different NN models and to obtain parsimony the appropriate model is selected based on AIC. Although there is a vast number of econometric models for modelling conditional volatility presented and estimated in this paper, along with an econometric presentation of NN models, estimation of 100 different NN models with hidden neurons ranging from 3 to 10 and comparing models with different specifications is econometrically unjustified.

Bildirici and Ersin [23] propose a family of nonlinear GARCH models that incorporate fractional integration (FI) and asymmetric power (AP) properties to MS-GARCH processes. Moreover, they augment the MS-GARCH type models with NN to improve forecasting accuracy. Therefore, the proposed MS-ARMA-FIGARCH, APGARCH, and FIAPGARCH processes are further augmented with MLP, RBF, Recurrent NN, and Hybrid NN type NNs. The MS-ARMA-GARCH

family and MS-ARMA-GARCH-NN family are utilized for modelling the daily stock returns of the ISE Index. Forecast accuracy is evaluated with MAE, MSE, and RMSE error criteria and Diebold-Mariano test for predictive accuracy. They conclude that the FI and AP counterparts of MS-GARCH model provided promising results, while the best results are obtained for their NN based models. Moreover, among the models analysed, the models MS-ARMA-FIAPGARCH-HNN and MS-ARMA-FIAPGARCH-RNN provided the best forecast performances over the single regime GARCH models and over the MS-GARCH model. Parameters of NN models are not explained econometrically, although NN are regarded as econometric model instead of nonparametric model.

Mantri et al [24] apply different methods, i.e. GARCH, EGARCH, GJR-GARCH, IGARCH and NN models for calculating the volatilities of Indian stock markets. Two networks are presented: single input (low index) single output (high index level) and multiple inputs (open, high and low index level) single output (close index level). The data from 1995 to 2008 of BSE Sensex and NSE Nifty indices are used to calculate the volatilities. The authors conclude that the MISO-NN model should be used instead of SISO-NN model and that there is no difference in the volatilities of Sensex and Nifty estimated under the GARCH, EGARCH, GJR-GARCH, IGARCH and NN models.

In their later paper Mantri et al. [25] focused on the problem of estimation of volatility of Indian Stock market. The paper begins with volatility calculation by ARCH and GARCH models of financial computation up to lag 3. The results are compared to NN model using R2. It can be concluded that NN can be used as a best choice for measuring the volatility of stock market. These papers provide no information about the particular NN model used, NN is not explained as econometric model, and therefore the papers are not suitable for deciding on suitability of the models.

Bektipratiwi and Irawan [26] propose an alternative forecasting model based on the combinations between RBF and EGARCH model to model stock returns of Bank Rakyat Indonesia Tbk for the period from 2003 to 2011. They use RBF to model the conditional mean and EGARCH to model the conditional volatility and propose a regression approach to estimate the weights and the parameters of EGARCH using maximum likelihood estimator. The relevant explanatory variables are chosen based on its contribution of giving greater reduction in the in-sample forecast errors. They selected 11 inputs for RBF model, and 5 hidden neurons based on trial and error procedure. Based on SIGN test, the best forecast is obtained by RBF-EGARCH model for 100 steps ahead.

All of the above researches combine GARCH-type and NN models by adding the NN structure to existing GARCH-type models in search of the suitable model for forecasting conditional variance of stock returns. The proposed methodology is empirically tested on developed markets, however not on developing capital markets of Central and Eastern Europe. Because of the uniqueness of these emerging capital markets, it is important to test features of proposed methodology in this particular segment. Moreover, some of the papers use NN as nonparametric estimation technique, neglecting the interpretability of parameters which could be obtained by using NN as econometric tool. In this paper NN will be observed as semiparametric approach combining the flexibility of nonparametric methods and the interpretability of parameters of parametric methods. Another contribution of the paper is in a priori specified structure of NN models in order to be comparable to the GARCH-type models and the estimation of additional parameter which was not estimated before.

3. DATA AND METHODOLOGY

The most widespread approach to volatility modelling consists of the GARCH model of Bollerslev [2] and its numerous extensions that can account for the volatility clustering and excess kurtosis found in financial time series. The accumulated evidences from empirical researches suggest that the volatility of financial markets can be appropriately captured by standard GARCH(1,1) model ([27]) since it gives satisfactory results with small number of parameters to estimate. According to Bollerslev [2] GARCH (1,1) can be defined as:

$$\begin{aligned} & r_t = \mu_t + \varepsilon_t \\ & \varepsilon_t = u_t \cdot \sqrt{\sigma_t^2} \\ & u_t : \ i.i.d.(0,1) \\ & \sigma_t^2 = \alpha + \beta \cdot \varepsilon_{t-1}^2 + \gamma \cdot \sigma_{t-1}^2 \end{aligned} \tag{1}$$

where μ_t is the conditional mean of return process $\{r_t\}$, while $\{\varepsilon_t\}$ is the innovation process with its multiplicative structure of identically and independently distributed random variables u_t. The last equation in (1) is conditional variance equation with GARCH(1,1) specification which means that variance of returns is conditioned on the information set I_{t-1} consisting of all relevant previous information up to and including period $t-1$. GARCH(1,1) model is covariance-stationary if and only if $\beta + \gamma < 1$ ([2]). In particular, GARCH(1,1) model usually indicates high persistence in the conditional variance, i.e. integrated behavior of the conditional variance when $\beta + \gamma = 1$ (IGARCH). The reason for the excessive GARCH forecasts in volatile periods may be the well-known high persistence of individual shocks in those forecasts. Relevant researches ([28]; [29]) show that this persistence may originate from structural changes in the variance process. High volatility persistence means that a long time period is needed for shocks in volatility to die out (mean reversion period).

Although GARCH models are the most popular and widely used in empirical researches and among practitioners due to their ability of describing the volatility clustering, excess kurtosis and fat-tailedness of the data, they cannot capture the asymmetric behavior of volatility. This means that negative shocks affect volatility quite differently than positive shocks. Therefore, different asymmetric models have been developed and used in empirical researches such as EGARCH, GJR-GARCH, TARCH, PGARCH, APGARCH among the others. However, the results from Engle and Ng [30] of Japanese stock returns suggest that Glosten, Jaganntahn and Runkle's [4] sign–ARCH model, usually called GJR model, shows the most potential in outperforming the traditional GARCH model. Moreover, in recent literature it also proved to capture the asymmetric behavior in data better that the other models ([20]). Therefore, GJR-GARCH model is considered, i.e. GJR-GARCH(1,1,1) is given by:

$$\begin{aligned} & \sigma_t^2 = \alpha + \beta \cdot \varepsilon_{t-1}^2 + \gamma \cdot \sigma_{t-1}^2 + \phi \cdot D_{t-1} \cdot \varepsilon_{t-1}^2 \\ & D_{t-1} = \begin{cases} 1 & if \quad \varepsilon_{t-1} < 0, \\ 0 & if \quad \varepsilon_{t-1} \geq 0. \end{cases} \end{aligned} \tag{2}$$

As can be seen from (2), GJR-GARCH model is just an augmentation of GARCH model that allows past negative unexpected returns to affect volatility differently than positive unexpected returns. When $\phi > 0$ negative shocks will have a larger impact on conditional variance. For GJR-GARCH stationarity condition is satisfied if $\beta + \gamma + \phi/2 < 1$.

An alternative solution to overcome the problems found for standard GARCH (1,1) model is to define appropriate neural network (NN), i.e. by extending the GJR-GARCH (1,1,1) model with NN model, significant improvements can be found.

The NN is an artificial intelligence method, which has recently received a great deal of attention in many fields of study. Usually NN can be seen as a nonparametric statistical procedure that uses the observed data to estimate the unknown function. A wide range of statistical and econometric models can be specified modifying the structure of the network, however NN often give better results. Empirical researches show that NN are successful in forecasting extremely volatile financial variables that are hard to predict with standard econometric methods such as: exchange rates ([7]), interest rates ([6]) and stocks ([8]). The most commonly used type of NN in empirical researches is multi-layer feed-forward neural networks (FNN).

The FNN forwards information from input layer to output layer through a number of hidden layers. Neurons in a current layer connect to neuron of the subsequent layer by weights and an activation function. In order to obtain weights backpropagation (BP) learning algorithm, which works by feeding the error back through the network, is mostly used. The weights are iteratively updated until there is no improvement in the error function. This process requires the derivative of the error function with respect to the network weights. The mean of squared error (MSE) is the conventional least square objective function in a NN, defined as mean of squared differences between the observed and fitted values of time series. The FNN with linear component can be written as:

$$\hat{y}_t = f\left(\phi_{co} + \sum_{i=1}^{p} \phi_{io} x_{t,i} + \sum_{h=1}^{q} \phi_{ho} g\left(\phi_{ch} + \sum_{i=1}^{p} \phi_{ih} x_{t,i}\right)\right) \tag{3}$$

where t is a time index, $\hat{y}_t$ is the output vector, $x_{t,i}$ is the input matrix with i variables, $f(\cdot)$ and $g(\cdot)$ are activation functions (usually linear and logistic respectively). Index c is the constant, i is the input, h is the hidden, and o is the output neuron. ϕ_{co} denotes the weight of the direct connection between the constant and output, ϕ_{io} denote the weights of direct connection from inputs to output, ϕ_{ch} denote the weights for the connections between the constant and hidden neurons. The weights ϕ_{ih} and ϕ_{ho} denote the weights for the connections between the inputs and hidden neurons and between the hidden neurons and output. NN with p inputs and q outputs has the abbreviation FNN(p,q).

However, the disadvantage of FNN is the problem of overfitting. It occurs due to the inclusion of multiple hidden layers or multiple neurons in hidden layer which, with existing theoretically based number of inputs (independent variables) and lagged outputs (dependent variables), increases the number of parameters to estimate. Therefore, in this paper NN are observed only as an extension to GJR-GARCH type model with the structure defined in advance to benefit from

parsimonious model in order to avoid the problem of overfitting. The GJR-GARCH-NN(1,1,1,1) as a nonlinear extension to GJR-GARCH model is defined as:

$$\begin{aligned}
&\sigma_t^2 = \alpha + \beta \cdot \varepsilon_{t-1}^2 + \gamma \cdot \sigma_{t-1}^2 + \phi \cdot D_{t-1} \cdot \varepsilon_{t-1}^2 + \xi \psi\left(z_t \lambda\right) \\
&D_{t-1} = \begin{cases} 1 & if \quad \varepsilon_{t-1} < 0, \\ 0 & if \quad \varepsilon_{t-1} \geq 0. \end{cases} \\
&\psi\left(z_t \lambda\right) = \frac{1}{1 + e^{\lambda z_{t-1}}} \\
&z_{t-1} = \frac{\varepsilon_{t-1} - E\left(\varepsilon\right)}{\sqrt{E\left(\varepsilon^2\right)}}.
\end{aligned} \tag{4}$$

where $\psi\left(z_t \lambda\right)$ specifies the logistic function in hidden unit of neural network with 1 hidden neuron, z_{t-1} provides a normalization of ε necessary to prepare the lagged unexpected returns as inputs into the nodes. All the data are transformed using the in-sample mean and variance. Donaldson and Kamstra [20] chose λ in advance from a uniform random number generator so they lie between -2 and 2 in order to achieve the identification of parameters ξ, and then parameters $\alpha, \beta, \gamma, \phi, \xi$ are estimated with maximum likelihood. In this paper λ are defined between -2 and 2, however they are estimated with maximum likelihood just as other parameters.

The data set consists of returns of the daily closing prices obtained from stock exchanges in period from January 2011 until September 2014 for selected European emerging markets, i.e. Bulgaria, Croatia, Czech, Romania, Slovakia and Slovenia. Data is obtained from Thomson Reuters database.

4. EMPIRICAL RESEARCH

In order to investigate GARCH-type models it is important to give an overview of the sample, i.e. descriptive statistics. From Table 1 can be seen that in observed period European emerging markets have negative expected returns. The lowest risk is observed in Slovakia and Slovenia and the highest risk in Czech Republic and Romania. Each distribution shows asymmetric behavior and leptokurtosis. Moreover, time series is not stationary since the variance of returns is time varying. Detailed results are omitted due to a lack of space. They are available from authors upon request.

Parameters for GJR-GARCH(1,1,1) model are estimated in SAS software using the maximum likelihood method and the results for selected markets with estimated parameters and the value of Log-Likelihood (LL) is given in Table 2. The results reveal that asymmetric behavior is statistically significant in all markets and since $\phi < 0$ the positive shocks will have a larger impact on conditional variance. In developed markets this parameter is usually positive indicating the opposite conclusions.

Table 1. Descriptive statistics of daily returns for selected markets

	N	Min	Max	μ	σ	α_3	α_4
BULGARIA	2177	-0,1136	0,0729	-0,00044	0,01307	-1,05	10,39
CROATIA	2177	-0,1076	0,1477	-0,00024	0,01273	0,14	18,43
CZECH	2177	-0,1618	0,1109	-0,00032	0,01521	-0,82	15,32
ROMANIA	2177	-0,1311	0,1056	-0,00005	0,01639	-0,50	8,64
SLOVAKIA	2177	-0,1481	0,1188	-0,00022	0,01153	-1,52	29,90
SLOVENIA	2177	-0,0843	0,0835	-0,00032	0,01189	-0,46	7,47

Table 1. Parameter estimates of GJR-GARCH(1,1,1) model with values of Log-Likelihood (LL)

	BULGARIA	CROATIA	CZECH	ROMANIA	SLOVAKIA	SLOVENIA
μ	-0.00007	-0.00021	-0.00008	0.000367*	5.36E-06	-0.00014
α	7.14E-06***	4.56E-07***	4.42E-03***	4.27E-06***	0.000028***	0.000012***
β	0.307146***	0.116087***	0.171614***	0.190021***	0.078577***	0.279281***
γ	0.703246***	0.912962***	0.846355***	0.824793***	0.797184***	0.697366***
ϕ	-0.08503**	-0.05222***	-0.08193***	-0.04265*	-0.09016***	-0.14047***
LL	6970.803	7227.855	6611.161	6470.448	6649.694	6945.997

Note: Parameter estimates are significant at 1% (***), 5% (**) and 10% (*) significance level

Parameters for GJR-GARCH-NN(1,1,1,1) model are estimated in SAS software using the maximum likelihood method and the results for selected markets with estimated parameters and the value of Log-Likelihood (LL) is given in Table 3. This model has two additional parameters to estimate: ξ and λ. Parameter ξ is in each market positive and statistically significant. Moreover, the Log-Likelihood is in GJR-GARCH-NN model larger than in simpler model. All these findings lead to a conclusion that extending the GJR-GARCH with the NN model, i.e. adding the nonlinearity in the model, is statistically significant and improves the models' fit

Table 3. Parameter estimates of GJR-GARCH-NN(1,1,1,1) model with values of Log-Likelihood (LL)

	BULGARIA	CROATIA	CZECH	ROMANIA	SLOVAKIA	SLOVENIA
μ	-0.00007	-0.00018	-0.00011	0.000391	-0.0002	-0.00015
α	-0.03772***	-0.03771***	-0.03772***	-0.03761***	-0.03787***	-0.03776***
β	0.311529***	0.12463***	0.151603***	0.19557***	0.364962***	0.349813***
γ	0.703012***	0.911607***	0.848221***	0.830741***	0.6018284***	0.697389***
ϕ	-0.09305***	-0.06524***	-0.04615	-0.06322***	-0.27835***	-0.27628***
ξ	0.075462***	0.075429***	0.075448***	0.075225***	0.075971***	0.075546***
λ	0.000028	0.000039	-0.00021	0.000137	0.000798*	0.000519***
LL	6970.826	7228.358	6612.326	6470.998	6657.298	6950.291

Note: Parameter estimates are significant at 1% (***), 5% (**) and 10% (*) significance level

5. CONCLUSIONS

Modelling volatility, i.e. returns fluctuations, is in the main focus of the paper. This research begins with the most widespread approach to volatility modelling, i.e. GARCH(1,1) model. Due to its disadvantage in capturing the asymmetric behavior GJR-GARCH(1,1,1) model is introduced. However, both models fail to model nonlinearity in data and, therefore NN model as an extension to GJR-GARCH model is defined, i.e. parsimonious GJR-GARCH-NN model. This paper estimates the parameters of both simple and extended GJR-GARCH model and compares these models using data for selected European emerging markets. Moreover, NN are presented as an econometric tool. Results of this paper confirm conclusions of previous researches about superiority of NN versus other linear and nonlinear models. However, they are still a challenge for the researchers in order to improve their performances in forecasting conditional variance of stock returns and time series in general. The out-of-sample predictive performance, inclusion of more hidden neurons or other architectures, the use of different algorithms in the network training, open space for future work and further studies.

ACKNOWLEDGEMENTS

This work has been fully supported by Croatian Science Foundation under the project "Volatility measurement, modeling and forecasting" (5199).

REFERENCES

[1] Engle, R.F. (1982) "Autoregressive conditional heteroscedasticity with estimates of the variance of UK inflation", Econometrica, Vol. 41, pp. 135-155.

[2] Bollerslev, T. (1986) "Generalized autoregressive conditional heteroscedasticity", Journal of Econometrics, Vol. 31, pp. 307-327.

[3] Diebold, F. (1986) "Comment on modelling the persistence of conditional variances," Econometric Reviews, Vol. 5, pp. 51–56.

[4] Glosten, L.R., Jagannathan, R. & Runkle, D.E. (1993) "On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks", The Journal of Finance, Vol. 48, No. 5 (Dec., 1993), pp. 1779-1801.

[5] Balkin, S.D. (1997) "Using Recurrent Neural Networks for Time Series Forecasting", Working Paper Series number 97-11, International Symposium on Forecasting, Barbados.

[6] Täppinen, J. (1998) "Interest rate forecasting with neural networks", Government Institute for Economic Research, Vatt-Discussion Papers, 170.

[7] Dunis, C.L. & Williams, M. (2002) "Modelling and trading the euro/US dollar exchange rate: Do neural networks perform better?", Journal of Derivatives & Hedge Funds, Vol. 8, No. 3, pp. 211-239.

[8] Zekić-Sušac, M. and Kliček, B. (2002) "A Nonlinear Strategy of Selecting NN Architectures for Stock Return Predictions", Finance, Proceedings from the 50th Anniversary Financial Conference Svishtov, Bulgaria, 11-12 April, Svishtov, Veliko Tarnovo, Bulgaria: ABAGAR, pp. 325-355.

[9] Tal, B. (2003) "Background Information on our Neural Network-Based System of Leading Indicators", CBIC World Markets, Economics & Strategy

[10] Ghiassi, M., Saidane, H. & Zimbra, D.K. (2005) "A dynamic artificial neural network model for forecasting time series events", International Journal of Forecasting, Vol. 21, pp. 341-362.

[11] Medeiros, M.C., Teräsvirta, T. & Rech, G. (2006) "Building Neural Network Models for Time Series: A Statistical Approach", Journal od Forecasting, No. 25, pp. 49-75.

[12] Kuan, C.-M. & White, H. (2007) "Artificial neural networks: an econometric perspective", Econometric Reviews, Vol. 13, pp. 1-92.

[13] Teräsvirta, T., Tjøstheim, D. & Granger, C.W.J. (2008) Modelling nonlinear economic time series, Advanced texts in econometrics, Oxford, New York, Oxford University Press

[14] Gonzales, S. (2000) "Neural Networks for Macroeconomic Forecasting: A Complementary Approach to Linear Regression Models", Working Paper 2000-07.

[15] Hwarng, H.B. (2001) "Insights into neural-network forecasting of time series corresponding to ARMA(p,q) structures", Omega, Vol. 29, pp. 273-289.

[16] Zhang, G.P. (2003) "Time series forecasting using hybrid ARIMA and neural network model", Neurocomputing, Vol. 50, pp. 159-175.

[17] Aminian, F., Suarez, E.D., Aminian, M. & Walz, D.T. (2006) "Forecasting Economic Data with Neural Networks", Computational Economics, Vol. 28, pp. 71-88.

[18] Lawrence, S., Giles, C.L. & Tsoi, A.C. (1997) "Lessons in Neural Network Training: Overfitting May be Harder than Expected", Proceedings of the Fourteenth National Conference on Artificial Intelligence, AAAI-97, pp. 540-545.

[19] Franses, P.H. & van Dijk, D. (2003) Nonlinear Time Series Models in Empirical Finance, Cambridge University Press.

[20] Donaldson, R.G. & Kamstra, M. (1997) "An Artificial Neural Network - GARCH Model for International Stock Return Volatility", Journal of Empirical Finance, Vol. 4, No. 1, pp. 17-46.

[21] Bildirici, M. & Ersin, Ö.Ö. (2009) "Improving forecasts of GARCH family models with the artificial neural networks: An application to the daily returns in Istanbul Stock Exchange", Expert Systems with Applications, No. 36, pp. 7355-7362

[22] Bildirici, M. & Ersin, Ö.Ö. (2012) "Nonlinear volatility models in economics: smooth transition and neural network augmented GARCH, APGARCH, FIGARCH and FIAPGARCH models", MPRA Paper No. 40330

[23] Bildirici, M. & Ersin, Ö.Ö. (2014) "Modelling Markov Switching ARMA-GARCH Neural Networks Models and an Application to Forecasting Stock Returns", Hindawi Publishing Corporation: The Scientific World Journal, Vol. 2014, Article ID 497941, 21 pages

[24] Mantri, J.K., Gahan, P. & Nayak, B.B. (2010) "Artificial Neural Networks – An Application to Stock Market Volatility", International Journal of Engineering Science and Technology, Vol. 2, No. 5, pp. 1451-1460.

[25] Mantri, J.K., Mohanty, D. &Nayak, B.B. (2012) "Design Neural Network for Stock Market Volatility: Accuracy Measurement", International Journal on Computer Technology & Applications, Vol. 3, No. 1, pp. 242-250.

[26] Bektipratiwi, A. & Irawan, M.I. (2011) "A RBF-EGARCH neural network model for time series forecasting", Proceedings of "The IceMATH 2011, Topic, pp. 1-8.

[27] Visković, J., Arnerić, J. &Rozga, A. (2014) "Volatility Switching between Two Regimes", World Academy of Science, Engineering and Technology, International Science Index 87, International Journal of Social, Management, Economics and Business Engineering, Vol. 8, No. 3, pp. 682 - 686.

[28] Lamoureux, C. & Lastrapes, W. (1990) "Persistence in variance, structural change, and the GARCH model", Journal of Business and Economic Statistics, Vol. 8, pp. 225–234.

[29] Wong, C.S. & Li, W.K. (2001) "On a mixture autoregressive conditional heteroscedastic model", Journal of American Statistical Association, Vol. 96, No. 455, pp. 982–995.

[30] Engle, R.F. & Ng, V.K. (1993) "Measuring and Testing the Impact of News on Volatility", The Journal of Finance, Vol. 48, No. 5 (Dec., 1993), pp. 1749-1778.

AUTHORS

Josip Arnerić

Assistant Professor, PhD. University of Zagreb, Faculty of Economics and Business Zagreb, Croatia. Scientific affiliation: econometric methods and models, financial time series and volatility, VAR models and cointegration, GARCH and MGARCH models, stochastic processes and risk management. Phone: 0038512383361. Fax: 0038512332618. E-mail: jarneric@efzg.hr

Tea Poklepović

Teaching Assistant, Doctoral student. University of Split, Faculty of Economics, Split, Croatia. Scientific affiliation: statistics and econometrics in business, finance and macroeconomics, especially econometric methods and models, time series and neural networks. Phone: 0038521430761. Fax: 0038521430701. E-mail: tpoklepo@efst.hr.

INTENTIONAL BLANK

MODIFIED VORTEX SEARCH ALGORITHM FOR REAL PARAMETER OPTIMIZATION

Berat Doğan

Department of Biomedical Engineering, Inonu University, Malatya, Turkey
berat.dogan@inonu.edu.tr

ABSTRACT

The Vortex Search (VS) algorithm is one of the recently proposed metaheuristic algorithms which was inspired from the vortical flow of the stirred fluids. Although the VS algorithm is shown to be a good candidate for the solution of certain optimization problems, it also has some drawbacks. In the VS algorithm, candidate solutions are generated around the current best solution by using a Gaussian distribution at each iteration pass. This provides simplicity to the algorithm but it also leads to some problems along. Especially, for the functions those have a number of local minimum points, to select a single point to generate candidate solutions leads the algorithm to being trapped into a local minimum point. Due to the adaptive step-size adjustment scheme used in the VS algorithm, the locality of the created candidate solutions is increased at each iteration pass. Therefore, if the algorithm cannot escape a local point as quickly as possible, it becomes much more difficult for the algorithm to escape from that point in the latter iterations. In this study, a modified Vortex Search algorithm (MVS) is proposed to overcome above mentioned drawback of the existing VS algorithm. In the MVS algorithm, the candidate solutions are generated around a number of points at each iteration pass. Computational results showed that with the help of this modification the global search ability of the existing VS algorithm is improved and the MVS algorithm outperformed the existing VS algorithm, PSO2011 and ABC algorithms for the benchmark numerical function set.

KEYWORDS

Metaheuristics, Numerical Function Optimization, Vortex Search Algorithm, Modified Vortex Search Algorithm.

1. INTRODUCTION

In the past two decades, a number of metaheuristic algorithms have been proposed to solve complex real-world optimization problems. Most of these algorithms are nature inspired methods and therefore mimic natural metaphors such as, evolution of species (GA [1] and DE [2-3]), annealing process (SA [4-5]), ant behaviour (ACO [6]), swarm behaviour (PSO [7] and ABC [8-9]) etc. These algorithms make few or no assumptions for the problem at hand and provide fast and robust solutions. Although, the solutions provided by metaheuristics may not be optimal solutions, they are highly preferred because of their simplicity and flexibility.

Despite the high number of available metaheuristics, developing new metaheuristic algorithms is still an active research area. In [10-15], a number of recently proposed metaheuristics can be

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 113–126, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60610

found. All of these metaheuristics have certain characteristics and thus each one may be more successful on a certain optimization problem when compared to the others. The Vortex Search (VS) algorithm [16] is one of these recently proposed metaheuristic algorithms which was inspired from the vortical flow of the stirred fluids. The search behaviour of the VS algorithm is modelled as a vortex pattern by using an adaptive step-size adjustment scheme. By this way, it is aimed to have a good balance between the explorative and exploitative behaviour of the search. The proposed VS algorithm was tested over 50 benchmark mathematical functions and the obtained results compared to the single-solution based (Simulated Annealing, SA and Pattern Search, PS) and population-based (Particle Swarm Optimization, PSO2011 and Artificial Bee Colony, ABC) algorithms. A Wilcoxon-Signed Rank Test was performed to measure the pair-wise statistical performances of the algorithms, the results of which indicated that the proposed VS algorithm outperforms the SA, PS and ABC algorithms while being competitive with the PSO2011 algorithm. Because of the simplicity of the proposed VS algorithm, a significant decrease in the computational time of the 50 benchmark numerical functions was also achieved when compared to the population-based algorithms. In some other studies [17-20], the VS algorithm has also been successfully used for the solution of some real-world optimization problems.

Although the proposed VS algorithm is a good candidate for the solution of optimization problems, it also has some drawbacks. In the VS algorithm, candidate solutions are generated around the current best solution by using a Gaussian distribution at each iteration pass. This provides simplicity to the algorithm but it also leads to some problems along. Especially, for the functions those have a number of local minimum points, to select a single point to generate candidate solutions leads the algorithm to being trapped into a local minimum point. Due to the adaptive step-size adjustment scheme used in the VS algorithm, the locality of the created candidate solutions is increased at each iteration pass. Therefore, if the algorithm cannot escape a local point as quickly as possible, it becomes much more difficult for the algorithm to escape from that point in the latter iterations.

In this study, a modified Vortex Search algorithm (MVS) is proposed to overcome above mentioned drawback of the existing VS algorithm. In the MVS algorithm, the candidate solutions are generated around different points at each iteration pass. These points are iteratively updated during the search process, details of which are given in the following section. The MVS algorithm is tested with 7 benchmark functions that was used earlier in [16]. These 7 functions are selected from the benchmark set of 50 functions for which the VS algorithm trapped into the local minimum points. Because the SA and PS algorithms showed poor performances in [16], in this study these two algorithms are excluded and the results are compared to the results those obtained by the VS algorithm, PSO2011 and ABC algorithms. It is shown that, the MVS algorithm outperforms all of these algorithms and can successfully escape from the local minimum points of the functions that the VS algorithm was being trapped earlier.

The remaining part of this paper is organized as follows. In the following section, first a brief description of the VS algorithm is given. Then, the modification performed on the VS algorithm is detailed and the MVS algorithm is introduced. Section 3 covers the experimental results and discussion. Finally, Section 4 concludes the work.

2. METHODOLOGY

2.1. A Brief Description of the Vortex Search Algorithm

Let us consider a two-dimensional optimization problem. In a two dimensional space a vortex pattern can be modelled by a number of nested circles. Here, the outer (largest) circle of the vortex is first centered on the search space, where the initial center can be calculated using Eq. 1.

$$\mu_0 = \frac{upperlimit + lowerlimit}{2} \tag{1}$$

In Eq.1, *upperlimit* and *lowerlimit* are $d \times 1$ vectors that define the bound constraints of the problem in d dimensional space. Then, a number of neighbor solutions $C_t(s)$, (t represents the iteration index and initially $t = 0$) are randomly generated around the initial center μ_0 in the d -dimensional space by using a Gaussian distribution. Here, $C_0(s) = \{s_1, s_2, ..., s_k\}$ $k = 1,2,...,n$ represents the solutions, and n represents the total number of candidate solutions. In Eq. 2, the general form of the multivariate Gaussian distribution is given.

$$p(x \mid \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right\} \tag{2}$$

In Eq.2, d represents the dimension, x is the $d \times 1$ vector of a random variable, μ is the $d \times 1$ vector of sample mean (center) and Σ is the covariance matrix. If the diagonal elements (variances) of the values of Σ are equal and if the off-diagonal elements (covariance) are zero (uncorrelated), then the resulting shape of the distribution will be spherical (which can be considered circular for a two-dimensional problem, as in our case). Thus, the value of Σ can be computed by using equal variances with zero covariance by using Eq. 3.

$$\Sigma = \sigma^2 \cdot [I]_{d \times d} \tag{3}$$

In Eq. 3, σ^2 represents the variance of the distribution and I represents the $d \times d$ identity matrix. The initial standard deviation (σ_0) of the distribution can be calculated by using Eq. 4.

$$\sigma_0 = \frac{\max(upperlimit) - \min(lowerlimit)}{2} \tag{4}$$

Here, σ_0 can also be considered as the initial radius (r_0) of the outer circle for a two dimensional optimization problem. Because a weak locality is required in the initial phases, r_0 is chosen to be a large value. Thus, a full coverage of the search space by the outer circle is provided in the initial step. This process provides a bird's-eye view for the problem at hand.

In the selection phase, a solution (which is the best one) $s' \in C_0(s)$ is selected and memorized from $C_0(s)$ to replace the current circle center μ_0. Prior to the selection phase, the candidate solutions must be ensured to be inside the search boundaries. For this purpose, the solutions that exceed the boundaries are shifted into the boundaries, as in Eq. 5.

$$s_k^i = \begin{cases} rand \cdot (upperlimit^i - lowerlimit^i) + lowerlimit^i, & s_k^i < lowerlimit^i \\ s_k^i, & lowerlimit^i \le s_k^i \le upperlimit^i \\ rand \cdot (upperlimit^i - lowerlimit^i) + lowerlimit^i, & s_k^i > upperlimit^i \end{cases} \quad (5)$$

In Eq.5, $k = 1,2,...n$ and $i = 1,2,...,d$ and $rand$ is a uniformly distributed random number. Next, the memorized best solution $s^{'}$ is assigned to be the center of the second circle (the inner one). In the generation phase of the second step, the effective radius (r_1) of this new circle is reduced, and then, a new set of solutions $C_1(s)$ is generated around the new center. Note that in the second step, the locality of the generated neighbors increased with the decreased radius. In the selection phase of the second step, the new set of solutions $C_1(s)$ is evaluated to select a solution $s^{'} \in C_1(s)$. If the selected solution is better than the best solution found so far, then this solution is assigned to be the new best solution and it is memorized. Next, the center of the third circle is assigned to be the memorized best solution found so far. This process iterates until the termination condition is met. An illustrative sketch of the process is given in Figure 1. In this manner, once the algorithm is terminated, the resulting pattern appears as a vortex-like structure, where the center of the smallest circle is the optimum point found by the algorithm. A representative pattern is sketched in Figure 2 for a two-dimensional optimization problem for which the upper and lower limits are between the [-10,10] interval. A description of the VS algorithm is also provided in Figure 3.

The radius decrement process given in Figure 3 can be considered as a type of adaptive step-size adjustment process which has critical importance on the performance of the VS algorithm. This process should be performed in such a way that allows the algorithm to behave in an explorative manner in the initial steps and in an exploitative manner in the latter steps. To achieve this type of process, the value of the radius must be tuned properly during the search process. In the VS algorithm, the inverse incomplete gamma function is used to decrease the value of the radius during each iteration pass.

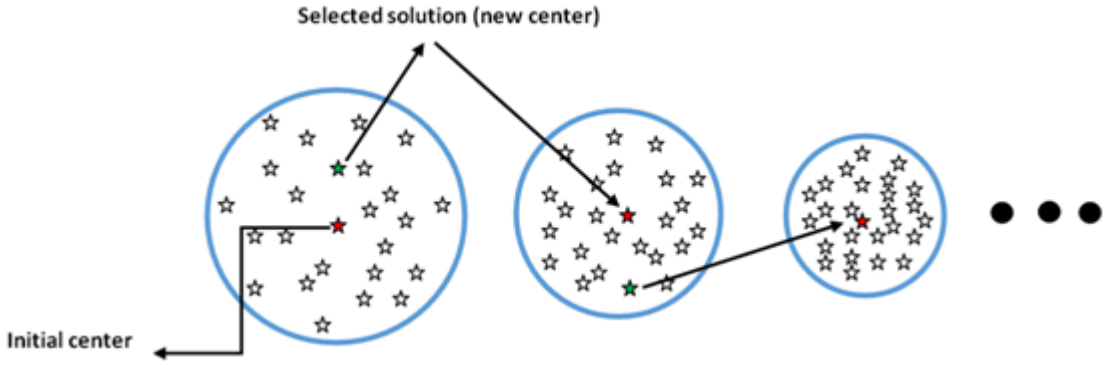


Figure 1. An illustrative sketch of the search process

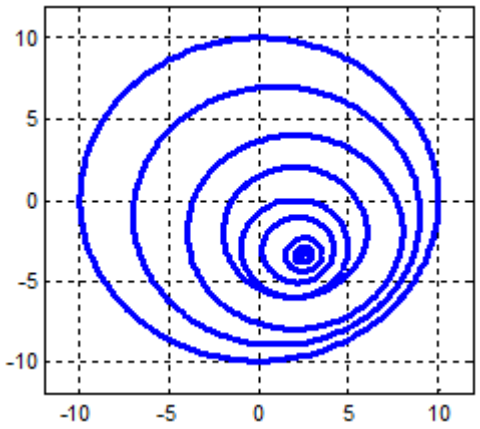


Figure 2. A representative pattern showing the search boundaries (circles) of the VS algorithm after a search process, which has a vortex-like structure.

The incomplete gamma function given in Eq. 6 most commonly arises in probability theory, particularly in those applications involving the chi-square distribution [21].

$$\gamma(x,a)=\int_0^x e^{-t}t^{a-1}dt \quad a>0 \tag{6}$$

In Eq.6, $a>0$ is known as the shape parameter and $x\geq 0$ is a random variable. In conjunction with the incomplete gamma function, its complementary $\Gamma(x,a)$ is usually also introduced (Eq. 7).

$$\Gamma(x,a)=\int_x^\infty e^{-t}t^{a-1}dt \quad a>0 \tag{7}$$

Thus, it follows that,

$$\gamma(x,a)+\Gamma(x,a)=\Gamma(a) \tag{8}$$

where $\Gamma(a)$ is known as the gamma function. There exist many studies in the literature on different proposed methods for the numerical calculation of the incomplete gamma function [22-24]. MATLAB® also provides some tools for the calculation of the inverse incomplete gamma (*gammaincinv*) function. The inverse incomplete gamma function (*gammaincinv*), computes the inverse of the incomplete gamma function with respect to the integration limit x and represented as *gammaincinv(x,a)* in MATLAB®.

In Figure 4, the inverse incomplete gamma function is plotted for $x=0.1$ and $a\in[0,1]$. Here, for our case the parameter a of the inverse incomplete gamma function defines the resolution of the search. By equally sampling a values within $[0,1]$ interval at a certain step size, the resolution of the search can be adjusted. For this purpose, at each iteration, a value of a is computed by using the Eq.9

$$a_t=a_0-\frac{t}{MaxItr} \tag{9}$$

Inputs: Initial center μ_0 is calculated by using Eq. 1
Initial radius r_0 (or the standard deviation, σ_0) is computed by using Eq. 10
Fitness of the best solution found so far $f(s_{best}) = \inf$
$t = 0$;
Repeat
/* Generate candidate solutions by using Gaussian distribution around the center μ_t with a standard deviation (radius) r_t*/
Generate($C_t(s)$) ;
If exceeded, then shift the $C_t(s)$ values into the boundaries as in Eq.5
/* Select the best solution from $C_t(s)$ to replace the current center μ_t */
$s^{'}$ = Select($C_t(s)$) ;
if $f(s^{'}) < f(s_{best})$
$s_{best} = s^{'}$
$f(s_{best}) = f(s^{'})$
else
keep the best solution found so far s_{best}
end
/* Center is always shifted to the best solution found so far */
$\mu_{t+1} = s_{best}$
/* Decrease the standard deviation (radius) for the next iteration */
r_{t+1} = Decrease(r_t)
$t = t + 1$;
Until the maximum number of iterations is reached
Output: Best solution found so far s_{best}

Figure 3. A description of the VS algorithm

where a_0 is selected as $a_0 = 1$ to ensure a full coverage of the search space at the first iteration, t is the iteration index, and $MaxItr$ represents the maximum number of iterations.

Let us consider an optimization problem defined within the [-10,10] region. The initial radius r_0 can be calculated with Eq. 10. Because $a_0 = 1$, the resulting function value is $(1/x) \cdot gammaincinv(x, a_0) \approx 1$, which means $r_0 \approx \sigma_0$ as indicated before.

$$r_0 = \sigma_0 \cdot (1/x) \cdot gammaincinv(x, a_0) \tag{10}$$

By means of Eq.4, the initial radius value r_0 can be calculated as $r_0 \approx 10$. In Eq.11, a general formula is also given to obtain the value of the radius at each iteration pass.

$$r_t = \sigma_0 \cdot (1/x) \cdot gammaincinv(x, a_t) \tag{11}$$

Here, t represents the iteration index.

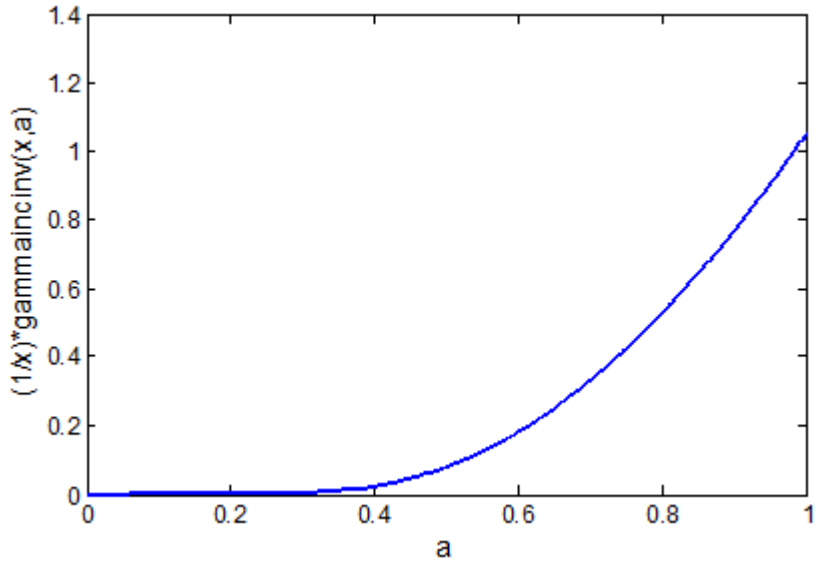


Figure 4. $(1/x)\cdot gammaincin\, v(x,a)$ where $x = 0.1$ and $a \in [0,1]$

2.2. The Modified Vortex Search Algorithm

The VS algorithm creates candidate solutions around a single point at each iteration pass. At the first iteration, this point is the initial center μ_0 which is determined with the upper and lower limits of the problem at hand while in the latter iterations the center is shifted to the current best position found so far. As mentioned before, this mechanism leads the VS algorithm to being trapped into local minimum points for a number of functions.

To overcome above mentioned drawback, in this study a modified VS algorithm (MVS) is proposed. In the MVS algorithm, candidate solutions are generated around multiple centers at each iterations pass. The search behavior of the MVS algorithm can be thought as a number of parallel vortices that have different centers at each iteration pass. Initially, the centers of these multiple vortices are selected as in the VS algorithm. Let us consider, the total number of centers (or vortices) to be represented by m. Let us say, $M_t(\mu)$ represents the matrix that stores the values of these m centers at each iteration pass and t represents the iteration index. Thus, initially $M_0(\mu) = \left\{\mu_0^1, \mu_0^2, ..., \mu_0^l\right\}$, $l = 1,2,...,m$ and initial positions of these centers are computed as in Eq. 12.

$$\mu_0^1 = \mu_0^2 = ... = \mu_0^l = \frac{upperlimit + lowerlimit}{2}, \quad l = 1,2,...,m \tag{12}$$

Next, a number of candidate solutions are generated with a Gaussian distribution around these initial centers by using the initial radius value r_0. In this case the total number of candidate solutions is again selected to be n. But note that, these n solutions are generated around m centers. Thus, one should select n/m solutions around each center.

Let us say, $CS_t^l(s) = \{s_1, s_2, ..., s_k\}$ $k = 1,2,...,n/m$ represents the subset of solutions generated around the center $l = 1,2,...,m$ for the iteration t. Then, the total solution set generated for the

iteration $t=0$ can be represented by $C_0(s)=\{CS_0^1, CS_0^2,...,CS_0^l\}, l=1,2,...,m$. In the selection phase, for each subset of solutions, a solution (which is the best one) $s_l^{'} \in CS_0^l(s)$ is selected. Prior to the selection phase it must be ensured that the candidate subsets of solutions are inside the search boundaries. For this purpose, the solutions that exceed the boundaries are shifted into the boundaries, as in Eq. 5. Let us say, the best solution of each subset is stored in a matrix $PBest_t(s^{'})$ at each iteration pass. Thus, for $t=0$, $PBest_0(s^{'})=\{s_1^{'}, s_2^{'},...,s_l^{'}\}, l=1,2,...,m$. Note that, the best solution of this matrix ($PBest_0(s^{'})$) is also the best solution of the total candidate solution set $C_0(s)$ for the current iteration, which is represented as Itr_{best}.

In the VS algorithm, at each iterations pass, the center is always shifted to the best solution found so far, s_{best}. However, in the MVS algorithm, there exist m centers which positions need to be updated for the next iteration. The most important difference between the VS and MVS algorithm arises from here. In the MVS algorithm, one of these centers is again shifted to the best solution found so far, s_{best}. But, the remaining $m-1$ centers are shifted to a new position determined by the best positions generated around the each center at the iteration t and the best position found so far, s_{best} as shown in Eq. 13.

$$\mu_t^l = s_l^{'} + rand \cdot (s_l^{'} + s_{best}) \tag{13}$$

In Eq. 13, *rand* is a uniformly distributed random number, $l=1,2,...,m-1$ and $s_l^{'} \in PBest_{t-1}(s^{'})$. Thus, for $t=1$, $M_1(\mu)=\{\mu_1^1, \mu_1^2,...,\mu_1^l\}$, $l=1,2,...,m-1$ is determined by using the $s_l^{'} \in PBest_0(s^{'})$ positions and the best position found so far, s_{best}. In Figure-5, an illustrative sketch of the center update process is given for a two-dimensional problem. In Figure 5, only one center is considered.

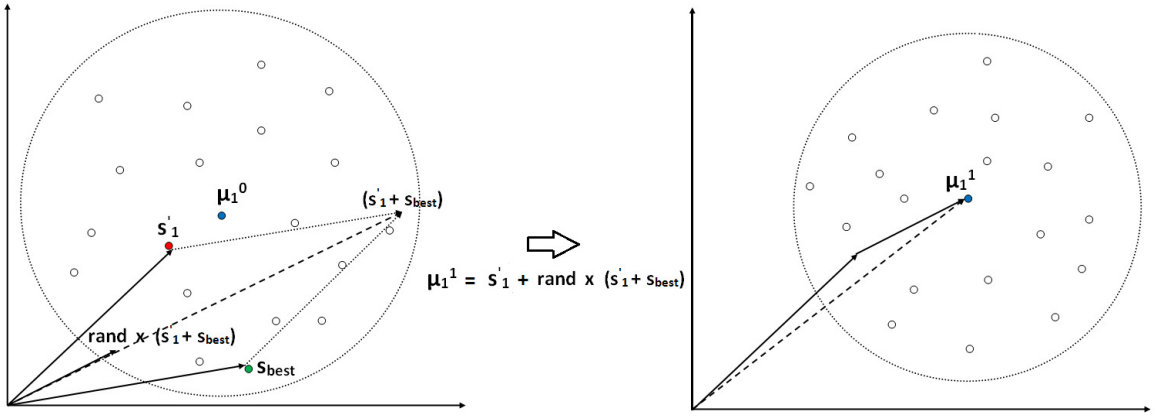


Figure 5. An illustrative sketch of the center updating process for the MVS algorithm (only one center is considered)

In the MVS algorithm, the radius decrement process is held totally in the same way as it is done in the VS algorithm. At each iteration pass, the radius is decreased by utilizing the inverse

incomplete gamma function and thus, the locality of the generated solutions is increased. In Figure 6, a description of the MVS algorithm is provided.

Inputs: Initial centers $M_0(\mu) = \{\mu_0^1, \mu_0^2, ..., \mu_0^l\}$, $l = 1,2,...,m$ is computed by using Eq. 12
Initial radius r_0 (or the standard deviation, σ_0) is computed by using Eq. 10
Fitness of the best solution found so far $f(s_{best}) = \inf$

$t = 0$;
Repeat
/* Generate candidate solution sets by using Gaussian distribution around the centers with a standard deviation (radius) r_t */
Generate($CS_t^l(s)$);
If exceeded, then shift the $CS_t^l(s)$ values into the boundaries as in Eq.5
/* Select the best solution from each subset $CS_t^l(s)$ to update the corresponding centers μ_t^l */
$s_l^{'}$ = Select($CS_t^l(s)$);
/* Store the best solution of each subset $CS_t^l(s)$ into the matrix $PBest_t(s^{'})$ */
$PBest_t(s^{'}) = Store(s_l^{'})$
/* Select the best solution Itr_{best} from the $PBest_t(s^{'})$ */
Itr_{best} = Select($PBest_t(s^{'})$)
if $f(Itr_{best}) < f(s_{best})$
$s_{best} = Itr_{best}$
$f(s_{best}) = f(Itr_{best})$
else
keep the best solution found so far s_{best}
end

Shift m-1 centers to their new positions as in Eq. 13
Shift one of the centers to the best solution found so far, s_{best}

/* Decrease the standard deviation (radius) for the next iteration */
r_{t+1} = Decrease(r_t)
$t = t + 1$;
Until the maximum number of iterations is reached
Output: Best solution found so far s_{best}

Figure 6. A description of the MVS algorithm

3. RESULTS

The proposed MVS algorithm is tested on 7 benchmark functions for which the VS algorithm was being trapped into a local minimum point. By using these functions, in this study, the

performance of the MVS algorithm is compared to the VS, PSO2011 and ABC algorithms. PSO2011 [25-26] is an extension of the standard PSO algorithm and the ABC algorithm is a well-known optimization algorithm which was inspired from the collective behaviours of honey bees.

The functions used in the experiments are listed in Table 1. For the formulations of the functions listed in Table 1, please refer to the reference [16].

3.1. Algorithm Settings

The ABC and PSO2011 algorithms are selected to have a population size of 50, which is also the number of neighborhood solutions of the proposed VS algorithm. The acceleration coefficients (c_1 and c_2) of the PSO2011 algorithm are both set to 1.8, and the inertia coefficient is set to 0.6, as in [27]. The *limit* value for the ABC algorithm is determined as *limit = SN * D,* where *SN* represents the number of food sources and *D* represents the dimension. VS algorithm does not have any additional parameters. Different from the VS algorithm, the MVS algorithm has the parameter m, which represents the total number of centers.

3.2. Experimental Results

For each algorithm, 30 different runs are performed, and the mean and the best values are recorded. The maximum number of iterations is selected to be 500,000. For the MATLAB® codes of the PSO2011, ABC, VS and MVS algorithms please refer to [25], [28], [29] and [30]. For each algorithm, all of the functions are run in parallel using a 32 core Intel® CPU 32 GB RAM workstation. For the first set of experiments, results are given in Table 2

Table 1. Benchmark function set that is used in the experiments

No	Function	Characteristics	Range	Dim.	Min.
F1	Powell	Unimodal Non-Separable	[-4,5]	24	0
F2	Rosenbrock	Unimodal Non-Separable	[-30, 30]	30	0
F3	Dixon-Price	Unimodal Non-Separable	[-10, 10]	30	0
F4	Rastrigin	Multimodal Separable	[-5.12, 5.12]	30	0
F5	Schwefel	Multimodal Separable	[-500, 500]	30	-12569.5
F6	Griewank	Multimodal Non-Separable	[-600, 600]	30	0
F7	Penalized	Multimodal Non-Separable	[-50, 50]	30	0

As shown in Table 2, for the MVS algorithm two different cases are considered. In the first case, the total number of candidate solutions is selected to be 50, which means 10 candidate solutions are generated around each center for $m = 5$. In this case, the MVS algorithm can avoid from the local minimum points of the functions which is not the case for the VS algorithm. However, poor sampling of the search space for this case (10 points around each center) may lead the MVS algorithm to show a correspondingly poor performance on the improvement of the found near optimal solutions (exploitation). Therefore, another case in which the total number of candidate solutions is selected to be 250 is considered for the MVS algorithm. In this case, 50 candidate solutions are generated around each center for $m = 5$. As can be shown in Table 2, the MVS algorithm with 250 candidate solutions performs better than the MVS algorithm with 50 candidate solutions.

In [31], authors stated that after a sufficient value for colony size, any increment in the value does not improve the performance of the ABC algorithm significantly. For the test problems carried out in [31] colony sizes of 10, 50 and 100 are used for the ABC algorithm. It is shown that although from 10 to 50 the performance of the ABC algorithm significantly increased, there is not any significant difference between the performances achieved by 50 and 100 colony sizes. Similarly, for the PSO algorithm it is reported that, PSO with different population sizes has almost the similar performance which means the performance of PSO is not sensitive to the population size [32]. Based on the above considerations, in this study a comparison of the MVS algorithm to the ABC and PSO2011 algorithms with a different population size is not performed. For the VS algorithm it is expected to achieve better exploitation ability with an increased number of candidate solutions. But the problem with the VS algorithm is with its global search ability rather than the local search ability for some of the functions listed above. Therefore, a comparison of the MVS (m = 5, n = 50) to VS algorithm with 50 candidate solutions is thought to be enough to show the improvement achieved by the modification performed on the VS algorithm.

In Figure 7, average computational time of 30 runs for 500,000 iterations is also provided for the MVS (m = 5, n = 50), MVS (m = 5, n = 250), VS, PSO2011 and ABC algorithms. As shown in this figure, the required computational time to perform 500,000 iterations with the MVS algorithm is slightly increased when compared to the VS algorithm. However, even for the MVS (m = 5, n = 250) algorithm the required computational time to perform 500,000 iterations is still lower than the PSO2011 and ABC algorithms.

Table 2. Statistical results of 30 runs obtained by PSO2011, ABC, VS and MVS algorithms (values < 10^{-16} are considered as 0).

No	Min.		MVS (m = 5, n = 50)	MVS (m = 5, n = 250)	VS	PSO2011	ABC
F1	0	Mean	7.59934E-09	3.88377E-10	1.43967E-05	2.04664E-07	9.09913E-05
		StdDev	4.14437E-08	1.27749E-09	2.27742E-06	1.21051E-08	1.42475E-05
		Best	1.1432E-16	0	5.71959E-06	1.72679E-07	5.23427E-05
F2	0	Mean	1.20813E-07	3.51659E-08	0.367860114	0.930212233	0.003535257
		StdDev	2.94163E-07	5.41004E-08	1.130879848	1.714978077	0.003314818
		Best	1.14463E-12	1.85577E-13	9.42587E-05	0	7.08757E-05
F3	0	Mean	0	0	0.666666667	0.666666667	1.91607E-15
		StdDev	0	0	7.68909E-16	4.38309E-16	2.55403E-16
		Best	0	0	0.666666667	0.666666667	1.1447E-15
F4	0	Mean	4.14483E-16	0	57.60799224	26.11016129	0
		StdDev	8.95296E-16	3.24317E-16	13.94980276	5.686650032	0
		Best	0	0	33.82857771	16.91429893	0
F5	0	Mean	-12569.48662	-12569.48662	-11283.05416	-8316.185447	-12569.48662
		StdDev	3.63798E-12	3.02118E-12	352.1869262	463.9606712	1.85009E-12
		Best	-12569.48662	-12569.48662	-11799.62928	-9466.201047	-12569.48662
F6	0	Mean	0	0	0.032798017	0.004761038	0
		StdDev	0	0	0.018570459	0.008047673	0
		Best	0	0	0.00739604	0	0
F7	-1	Mean	0	0	0.114662313	0.024187276	2.63417E-16
		StdDev	0	0	0.532276418	0.080213839	0
		Best	0	0	0	0	1.29727E-16

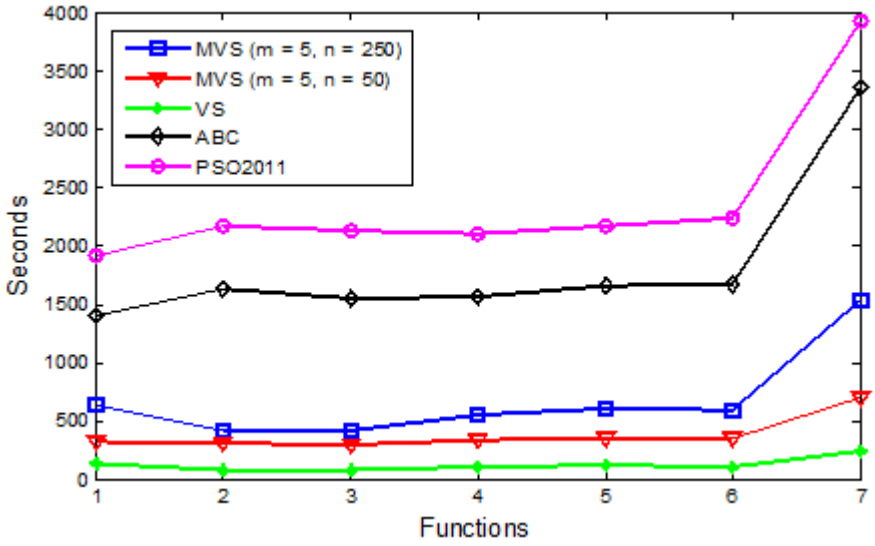


Figure 7. Average computational time of 30 runs for 50 benchmark functions (500,000 iterations)

4. CONCLUSIONS

This paper presents a modified VS algorithm in which the global search ability of the existing VS algorithm is improved. This is achieved by using multiple centers during the candidate solution generation phase of the algorithm at each iteration pass. In the VS algorithm, only one center is used for this purpose and this usually leads the algorithm to being trapped into a local minimum point for some of the benchmark functions. Computational experiments performed on the benchmark functions showed that, the MVS algorithm outperforms the VS, PSO211 and ABC algorithms and can successfully escapes from the local minimum points of the functions that the VS algorithm was being trapped earlier. Although the complexity of the existing VS algorithm is a bit increased with the performed modification, there is not any significant difference between the computational time of the modified VS algorithm and the existing VS algorithm.

In the future studies, the proposed MVS algorithm will be used for the solution of some real world optimization problems such as neural network optimization, optimum data partitioning, and analog circuit parameters optimization.

REFERENCES

[1] Holland J.H., Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, 1975

[2] Storn R., Price K., Differential evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces, Technical report, International Computer Science Institute, Berkley, 1995

[3] Storn R., Price K., Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341–359.

[4] Kirkpatrick S., Gelatt Jr C.D., Vecchi M.P., Optimization by Simulated Annealing, Science 220 (4598): 671–680, (1983).

[5] Černý V., Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm, Journal of Optimization Theory and Applications, 45: 41–51, (1985).

[6] Dorigo M., Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italy, 1992

[7] Kennedy J., Eberhart R.C., in: Particle Swarm Optimization, 1995 IEEE International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948

[8] Karaboga D., An idea based on honeybee swarm for numerical optimization, Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[9] Karaboga D., Basturk B., A. powerful, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm, Journal of Global Optimization 39 (3) (2007) 459–471.

[10] Civicioglu P., Backtracking Search Optimization Algorithm for numerical optimization problems, Applied Mathematics and Computation, Volume 219, Issue 15, 1 April 2013, Pages 8121-8144, ISSN 0096-3003

[11] Kashan A.H., A new metaheuristic for optimization: Optics inspired optimization (OIO), Computers & Operations Research, Volume 55, March 2015, Pages 99-125, ISSN 0305-0548

[12] Yang X.S., Flower pollination algorithm for global optimization, Unconventional computation and natural computation. Springer Berlin Heidelberg, 2012. 240-249

[13] Hajipour H., Khormuji H.B., and Rostami H., ODMA: a novel swarm-evolutionary metaheuristic optimizer inspired by open source development model and communities. Soft Computing (2014): 1-21.

[14] Yong L., Peng T., A multi-start central force optimization for global optimization, Applied Soft Computing, Volume 27, February 2015, Pages 92-98, ISSN 1568-4946

[15] Yu-Jun Z., Water wave optimization: A new nature-inspired metaheuristic, Computers & Operations Research, Volume 55, March 2015, Pages 1-11, ISSN 0305-0548

[16] Doğan B., Ölmez T., A new metaheuristic for numerical function optimization: Vortex Search algorithm, Information Sciences, Volume 293, 1 February 2015, Pages 125-145, ISSN 0020-0255

[17] Doğan B., Ölmez T., Vortex search algorithm for the analog active filter component selection problem, AEU - International Journal of Electronics and Communications, Volume 69, Issue 9, September 2015, Pages 1243-1253, ISSN 1434-8411

[18] Doğan, B., Yuksel, A., Analog filter group delay optimization using the Vortex Search algorithm, Signal Processing and Communications Applications Conference (SIU), 2015 23th , vol., no., pp.288,291, 16-19 May 2015

[19] Doğan B., Ölmez T., Modified Off-lattice AB Model for Protein Folding Problem Using the Vortex Search Algorithm, International Journal of Machine Learning and Computing vol. 5, no. 4, pp. 329-333, 2015.

[20] Doğan B., Ölmez T., Fuzzy clustering of ECG beats using a new metaheuristic approach, 2nd International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO), 7-9 April 2014, Granada, Spain.

[21] Andrews L.C., Special Functions of Mathematics for Engineers, SPIE Press, 1992

[22] Gautschi W., A note on the recursive calculation of incomplete gamma functions ACM Trans. Math. Software, 25 (1) (1999), pp. 101–107

[23] Winitzki S., Computing the incomplete Gamma function to arbitrary precision Computational Science and Its Applications – ICCSA 2003, of LNCS, Vol. 2667 Springer-Verlag, Berlin (2003), pp. 790–798

[24] Allasia, G., Besenghi R., Numerical calculation of incomplete gamma function by the trapezoidal rule, Numer. Math. (Numerische Mathematik) 50 (4):419{428, 1987

[25] Omran M.G.H., Clerc M., 2011, <http://www.particleswarm.info/>, accessed 25 February 2016

[26] Clerc M., "Standard Particle Swarm Optimization," Particle Swarm Central, Tech. Rep., 2012, http://clerc.maurice.free.fr/pso/SPSO_descriptions.pdf, accessed 25 February 2016

[27] Karaboga, D., Akay, B., A comparative study of Artificial Bee Colony algorithm, Applied Mathematics and Computation, Volume 214, Issue 1, 1 August 2009, Pages 108-132, ISSN 0096-3003.

[28] ABC algorithm, http://mf.erciyes.edu.tr/abc/, accessed 25 February 2016

[29] VS algorithm, http://web.itu.edu.tr/~bdogan/VortexSearch/VS.htm, accessed 25 February 2016

[30] MVS algorithm, http://web.itu.edu.tr/~bdogan/ModifiedVortexSearch/MVS.htm, accessed 25 February 2016

[31] Karaboga D., Basturk B., On the performance of artificial bee colony (abc) algorithm, Applied Soft Computing 8 (1) (2008) 687–697.

[32] Shi, Y., Eberhart R.C., Empirical study of particle swarm optimization, Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on. Vol. 3. IEEE, 1999.

AUTHORS

Dr. Berat Doğan received his BSc. degree in Electronics Engineering from Erciyes University, Turkey, 2006. He received his MSc. degree in Biomedical Engineering from Istanbul Technical University, Turkey, 2009. He received his PhD. in Electronics Engineering at Istanbul Technical University, Turkey, 2015. Between 2008-2009 he worked as a software engineer at Nortel Networks Netas Telecommunication Inc. Then, from 2009 to July 2015 he worked as a Research Assistant at Istanbul Technical University. Now he is working as an Assistant Professor at Inonu University, Malatya, Turkey. His research interests include optimization algorithms, pattern recognition, biomedical signal and image processing, and bioinformatics.

ANALYSIS OF RISING TUITION RATES IN THE UNITED STATES BASED ON CLUSTERING ANALYSIS AND REGRESSION MODELS

Long Cheng[1,2] and Chenyu You[1,2]

[1]Department of Electrical, Computer and Systems Engineering,
[2]Department of Mathematical Sciences,
Rensselaer Polytechnic Institute, Troy, NY, USA
dearlongcheng@gmail.com youc2@rpi.edu

ABSTRACT

Since higher education is one of the major driving forces for country development and social prosperity, and tuition plays a significant role in determining whether or not a person can afford to receive higher education, the rising tuition is a topic of big concern today. So it is essentially necessary to understand what factors affect the tuition and how they increase or decrease the tuition. Many existing studies on the rising tuition either lack large amounts of real data and proper quantitative models to support their conclusions, or are limited to focus on only a few factors that might affect the tuition, which fail to make a comprehensive analysis. In this paper, we explore a wide variety of factors that might affect the tuition growth rate by use of large amounts of authentic data and different quantitative methods such as clustering analysis and regression models.

KEYWORDS

Higher Education, Tuition Growth Rate, K-means Clustering, Linear Regression, Decision Tree, Random Effect

1. INTRODUCTION

There is no doubt that education is vital to the development of a country [1], especially considering now the United States has transferred from a labour-intensive economy to a knowledge-based economy [2]. Research study in [3] shows that college education is becoming increasingly expensive while the return on its investment is falling. It is believed that this trend will eventually affect how people view the importance of education to individuals in the future even though between 2000 and 2010, undergraduate tuition fee is increased by 3.4 percent each year on average at public four-year colleges and 2.8 percent at private institutions; both of them are on average 2.5 percent higher than the average annual rate of inflation. In addition, the second largest expense in a family is the expenditure for children's education, which might indicate that the increase in income will increase the demand in higher education and finally affect the tuition fee.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 127–144, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60611

The rising tuition fee plagues thousands of families every year [4], and the increase rate of tuition fee becomes higher and higher, which motivate us in this paper to explore what are the key factors that cause such high rate. Initially we think there are two main reasons for the rise in the tuition fee. One is the increase in university expenses. The adjusted cost at public universities was increased on average by 28 percent from 2000 to 2010. The other reason is the recession, which led a budget crisis. Education is one of the largest components of state budget. Hence, the government has to reduce the education expenditure. Since "early leaning" is critical to children's success in the future, the government decided to preserve the most essential programs, K-12 basic education, for the youngest citizens but cut more budgets in higher education to avoid budget shortage.

A fixed-effects model using cross-sectional and time-series date is proposed in [7] to explore how different racial groups react to the tuition fee. A method using multivariate regression analysis to investigate the relationship between enrollment rates and tuition fees is studied in [9] with the consideration of different races, enrolment rates and states. [10] studies how the change of the tuition fee affects personal decision on higher education and shows that even though both higher student scholarship and lower tuition fee would increase the students enrolment rate, the actual amount of money that students have to pay by themselves is a more important factor that affects their decision. [13] applies the investment theory and the consumption approach to study the demand for higher education with the consideration of the tuition fee. And the result shows that the demand for higher education react positively to family income increase and negatively to tuition fee increase. The effect of socioeconomic status of students on education opportunity and tuition fee is explored in [11]. The conclusion in [12] indicates the tuition fee has less impact than some other micro-econometric factors on college enrollment. It is also showed that the change of the tuition fee might have different impacts on different geographic locations, which are also affected by the tax rate [15].

Though some preliminary research on tuition fee analysis has been done in above papers, they fail to present a very detailed analysis using large amounts of raw data and considering a wide variety of factors. This paper explores many different factors, including both in-school ones and macro-economic ones, that might have impacts on tuition levels. And we find out that possible in-school factors that determine tuition price might be university ranking, class size, percentage of full-time faculty, college acceptance rate, classroom equipment, financial need, and the number of full-time students. In the macro-economic prospect, unemployment rate, price level, government policy and the average income level might make great effects on the rise of college tuition in the United States. Other possible factors that might contribute to the increment of tuition rates include geography and population.

2. DATA DESCRIPTION

A reliable dataset from National Center for Education Statistics IPEDS Analytics: Delta Cost Project Database [6] is used in this paper. This is a longitudinal database derived from IPEDS finance, enrollment, staffing, completions and student aid data for academic years 2000-2001 through 2009-10.

There are 950 variables used in the dataset to record information about tuition fee, state name, school name, employee number, and so on. All the variables are carefully reviewed and finally 30 variables are selected in this paper to perform the analysis. For example, considering the fact that

many graduate students are part-time students, whose tuition fees vary a lot even though they are in the same graduate program, so we decide to delete all data related with graduate students, and only focus on undergraduate students. Meanwhile, schools are divided into three categories: private school, public school with in-state tuitions, and public school with out-of-state tuitions, for the convenience of analysis. After the initial variable selection procedure, those variables are again inspected to see if they contain too many missing values. There are five variables (see Figure 1) that contain more than 40000 missing values, which are more than half of the total number (the total number is around 80000). Even though there are some existing methods to add the missing data, the accuracy of the analysis could be impaired if we use those methods to add so many missing data for those five variables. So we decide to delete those five variables and only use the other 25 variables. This paper will further explain how these variables are selected and processed in details in Part 3 ANALYSIS.

```
> miss
          academicyear                instname                   admssn              affiliate01              all_employees
                     0                       0                    50687                    70856                    33696
           conthoursug                 control               cpi_index          cpi_scalar_2010                  eandg01
                 52624                       0                        0                        0                    12057
faculty_instr_headcount fall_cohort_num_instate fall_cohort_num_outofstate fall_cohort_num_resunknown fall_total_undergrad
                 38284                   33666                    33669                    33680                    25055
             fte_count       grad_rate_150_n2yr       grad_rate_150_n4yr                  grant07               nettuition01
                 10899                   58432                    58369                    13420                    12354
   restricted_revenue              salarytotal                    state  tot_rev_w_auxother_sum       total_faculty_all
                 17748                   38284                        0                    11871                    32339
          tuition02_tf               tuition03               tuition03_tf          tuitionfee02_tf          tuitionfee03_tf
                 34029                   12453                    34020                    34232                    34232
```

Figure 1. Missing values of the selected 30 variables

3. ANALYSIS

This data set contains around 1000 schools' information. Since most students only care more about schools they are familiar with, our paper only focuses on those well-known universities. So we set up the minimum bounds of 25% quantile and 75% quantile SAT scores as 400 and 600, for both Math and Verbal sections. After this SAT score selection, there are 370 schools left.

In addition, 9-year data is used in this paper to analyze the trend of tuition rates. For schools with complete 9-year tuition information, we calculate the annual tuition growth rate using [(this year / the previous year)-1] for each year. For schools with only 8-year tuition information, we predict the annual tuition growth rate by [the square root of (the latter year/ the previous year)-1] for the gap year. For instance, if one school misses 2008 tuition information, we predict 2008 tuition growth rate by [square root of (tuition 2009/tuition 2007) -1]. Those schools with less than 8-year tuition data are deleted from the dataset. After above steps, the dataset now contains 9-year tuition information for about 300 universities in 42 states. Lastly, the employee/student ratio and faculty /student ratio are calculated using the raw data and added into the final dataset as two new variables.

For a clear presentation, tuition growth rates in the 42 states will be shown using figures according to three categories (private schools, public schools with in-state tuitions, and public schools with out-of-state tuitions). In these figures, x-axis stands for the year, and y-axis stands for the tuition growth rate. For each state, the 25% quantile, 50% quantile, and 75% quantile of

the tuition growth rates of all the schools belonging to the same category are calculated and shown in the following figures.

(1) Private schools

Figure 2 shows the tuition growth rates from 2002 to 2009 for private schools in 42 states. And we remove graphs of some states that don't have significant changes of tuition growth rates. In the end, there are 16 graphs left, which means that there are 16 states that have apparent trend changes in tuition growth rates.

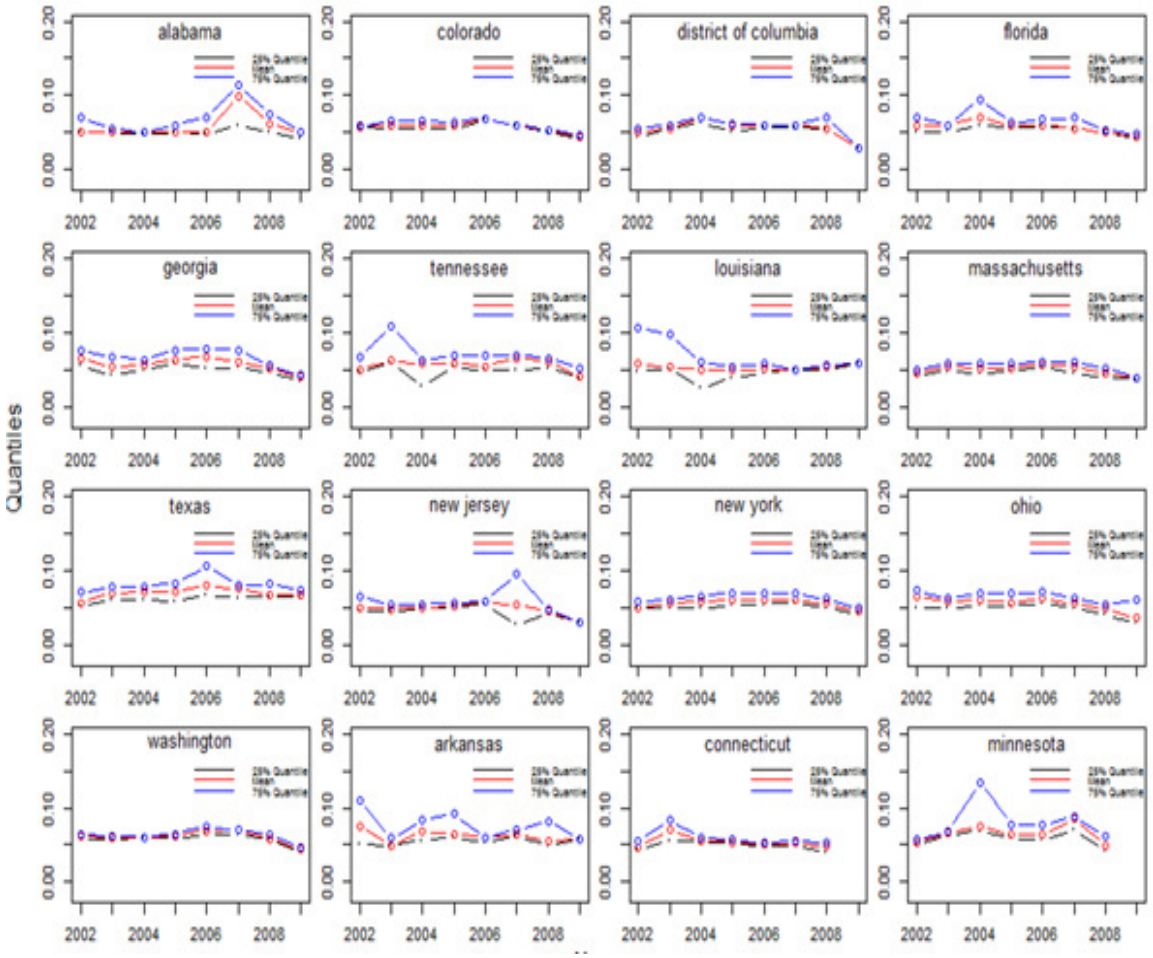


Figure 2. Tuition growth rates from 2002 to 2009 for private schools in 16 states

From this figure, it can be seen that in most academic years, the tuition growth rates change slowly. However, there are two academic years that have sharp changes. One is academic year 2003-2004, another one is 2008-2009. For almost all states, there is a decrease in tuition growth rates for 2008-2009. We guess that this is related with 2008 Financial Crisis, which leads to a deflation, making the tuition growth rate decrease. One thing needs to be noticed is that even though the tuition growth rate decreases for 2008-2009, the tuition fee still increases.

(2) Public schools with in-state tuition

Similarly, there are 12 states that have apparent trend changes in tuition growth rates, as shown in Figure 3.

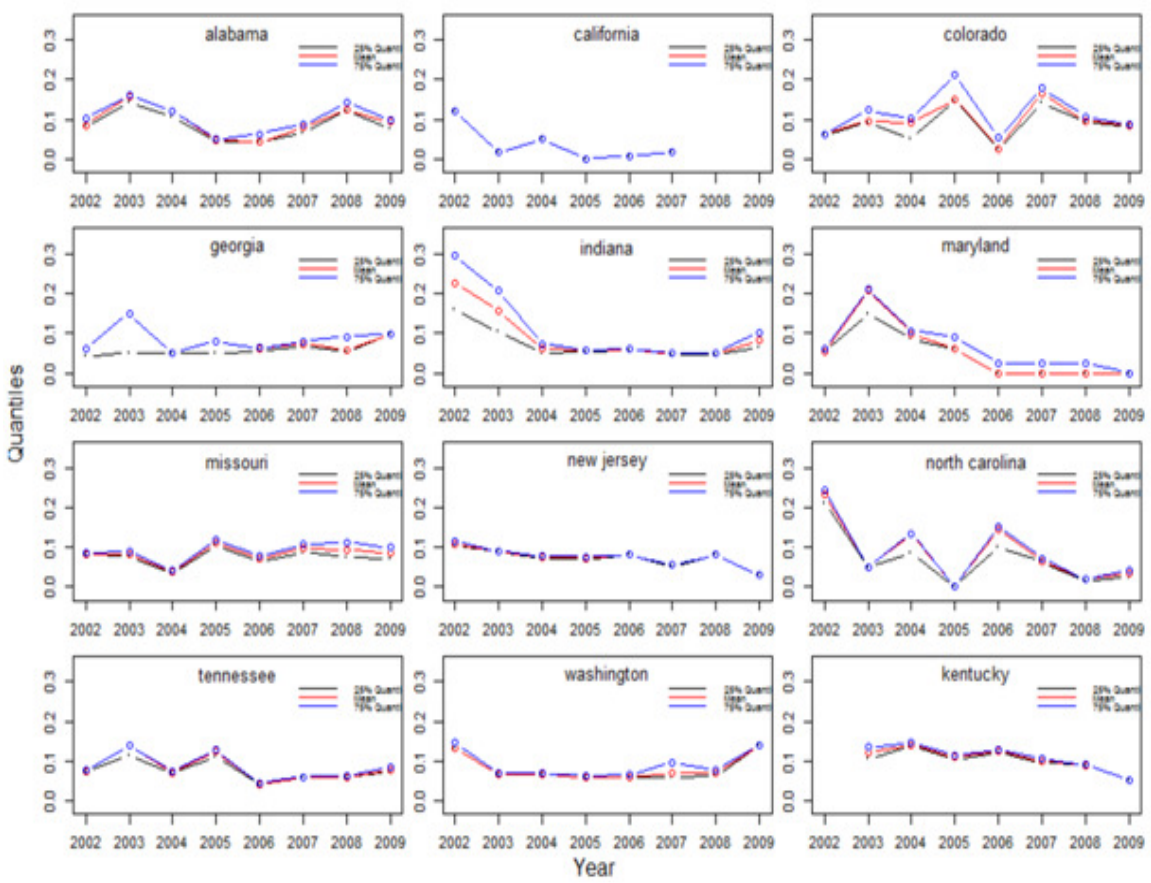


Figure 3. Tuition growth rates from 2002 to 2009 for public schools with in-state tuition in 12 states

Since all public schools belong to the UC system in California and all those schools have the same tuition growth rate, there is only one line in the graph for California.

To figure out the reason why Alabama, Colorado, Indiana, Maryland, and North Carolina have significant changes of tuition growth rates, the original dataset is checked and it is noticed that only a few schools belongs to this category in these five states. So changes of tuition growth rates in one school affect the results of the whole state a lot.

(3) Public schools with out-of-state tuition

Similarly, there are 14 states that have apparent trend changes in tuition growth rates, as shown in Figure 4. We find out that almost all these 14 states, except for Texas, have slow changes during those years.

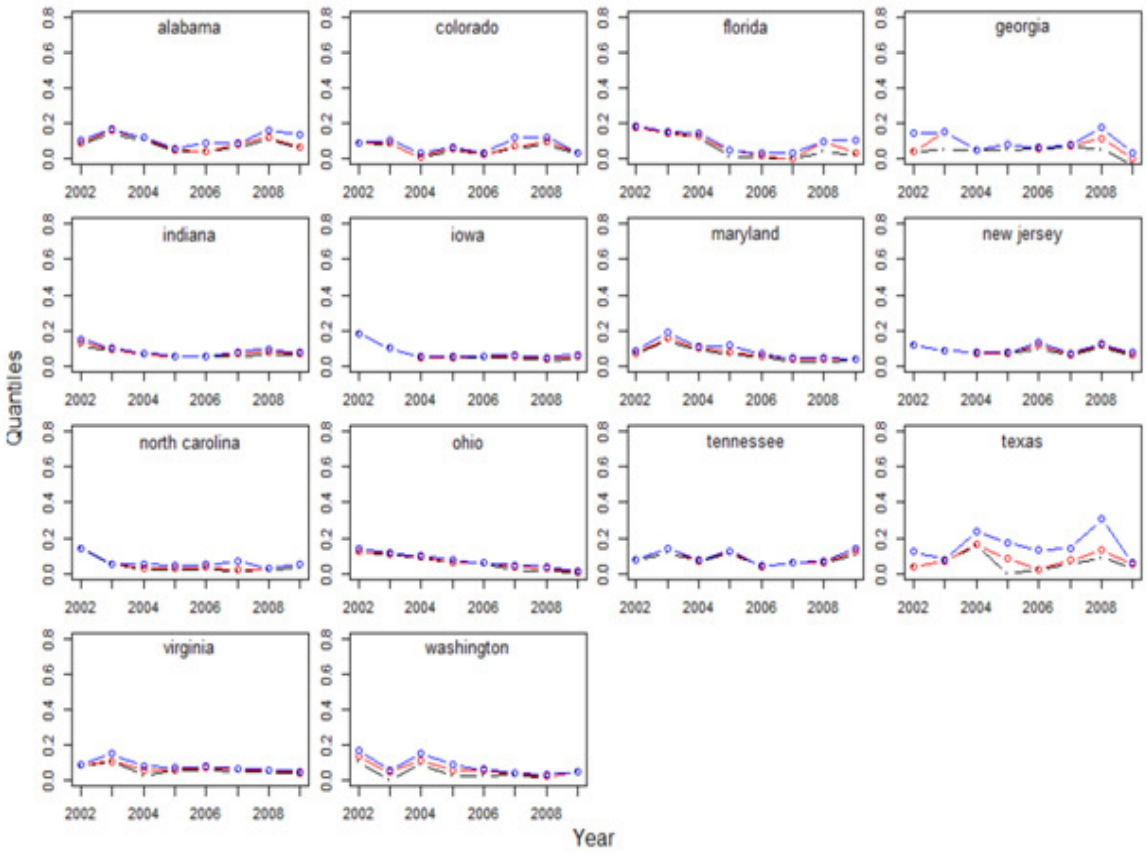


Figure 4. Tuition growth rates from 2002 to 2009 for public schools without-state tuition in 14 states

4. MODEL DEVELOPMENT

4.1 Clustering Analysis

K-means clustering is utilized in this paper to analyze the dataset based on four criteria: quantiles of all years, quantile in every single year, tuition growth rates, and tuition fees. For each of those criteria, different numbers of clusters are chosen. We also draw state maps to show the clustering results for each category for all years. And in all these maps, the smaller cluster number represents the lower tuition growth rate, and the larger cluster number represents the higher tuition growth rate. For instance, Cluster #1 represents for the lowest tuition growth rate.

4.1.1 Using Quantiles of All Years

(1) Choose the number of clusters

From Figure 5 and Figure 6, we decide to use 4 clusters for private schools, and 6 clusters for public schools with out-of-state tuition. There is no need to cluster public schools with in-state tuition, since the corresponding figure is only a horizontal line.

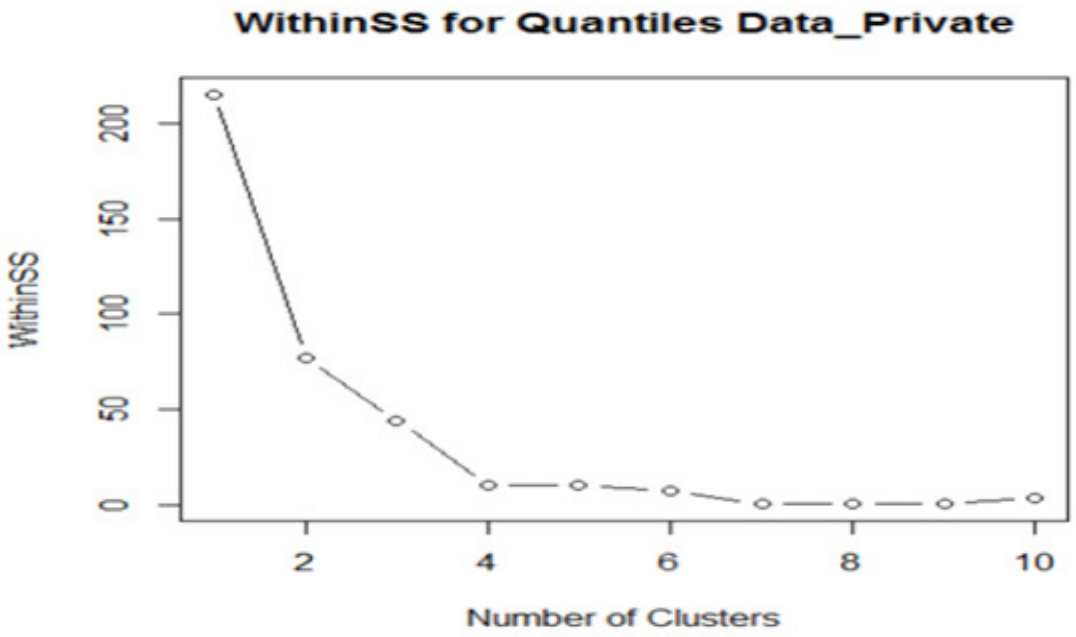


Figure 5. WithinSS for quantiles data (private schools)

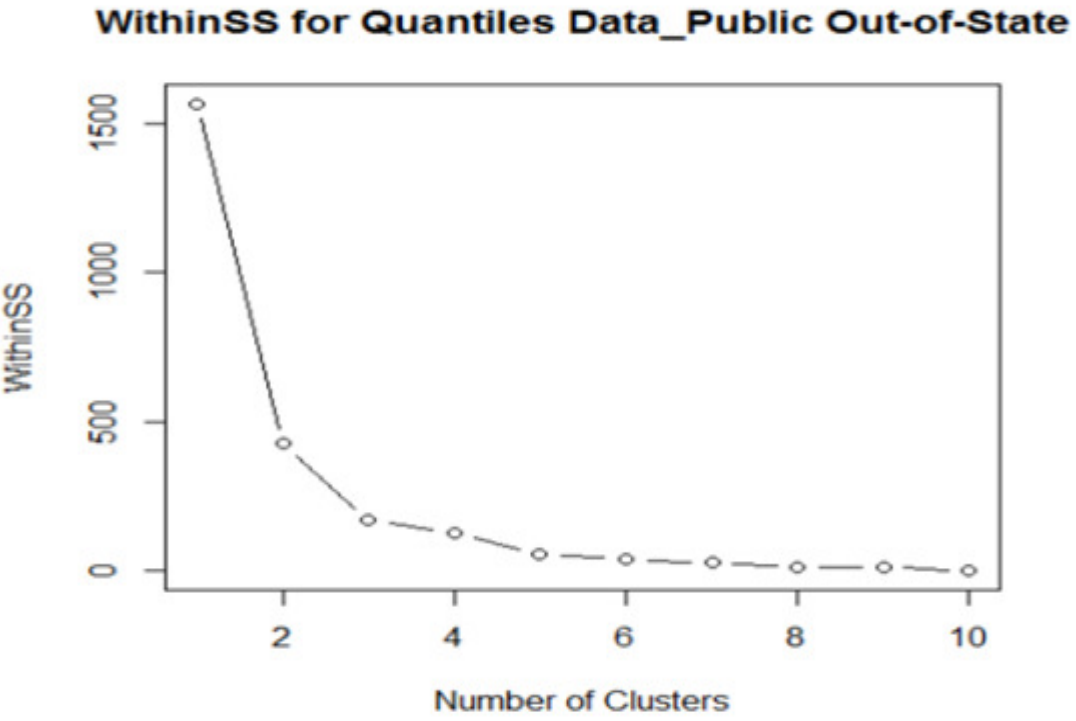


Figure 6. WithinSS for quantiles data (public schools with out-of-state tuition)

(2) Maps for private schools

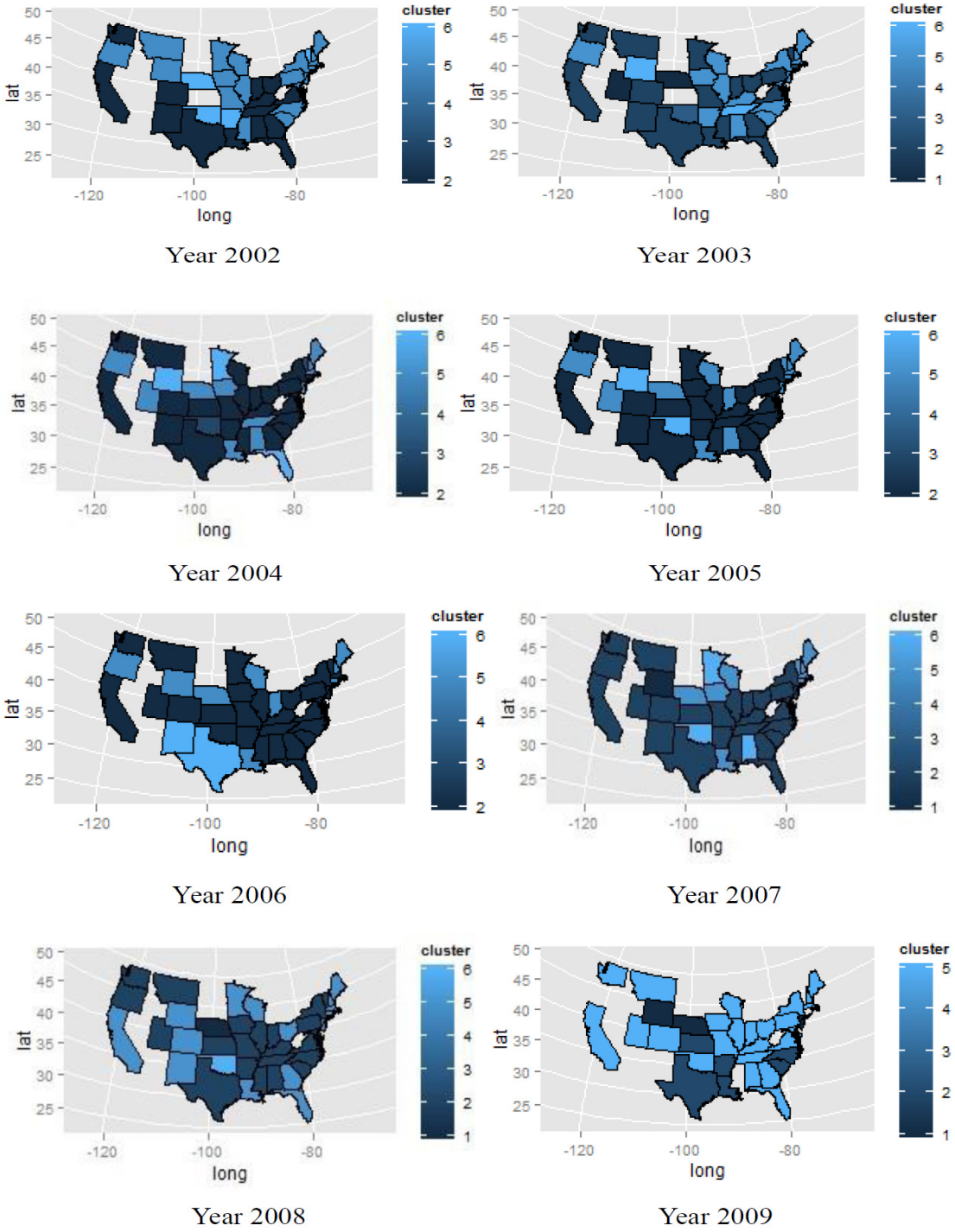


Figure 7. Clustering maps for private schools using quantiles for all years

(3) Maps for public schools with out-of-state tuition

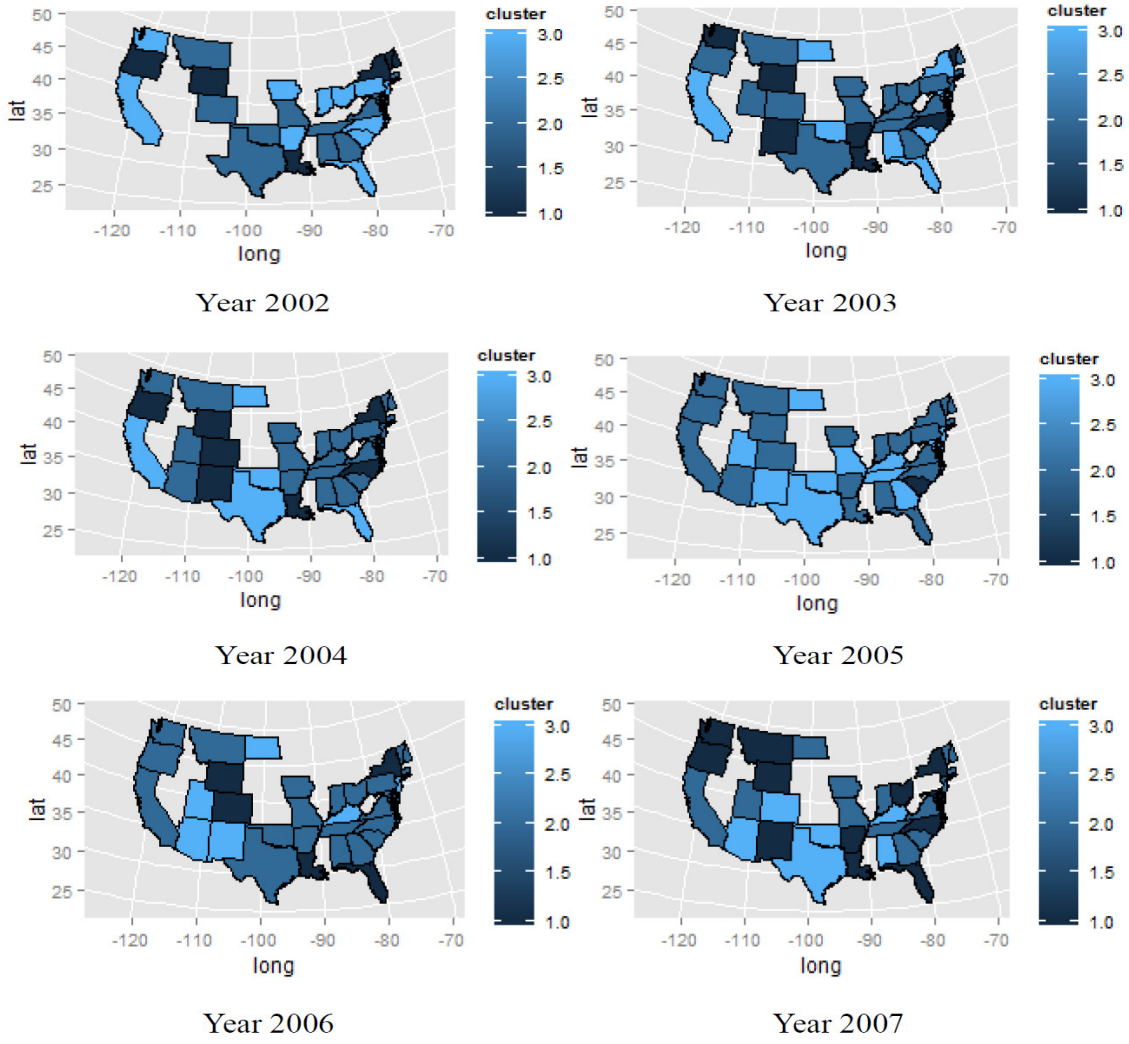
Even though there are 6 clusters in total for public schools with out-of-state tuition, there is only 1 cluster for each year. So we do not draw the figure. After checking the original data, we find out that this is because the number of public schools is too small.

4.1.2 Using Quantiles in Every Single Year

(1) Choose the number of clusters

Repeat the same procedure as the above section, we choose to use 3 clusters for both private schools and public schools without-state tuition.

(2) Maps for public schools with out-of-state tuition



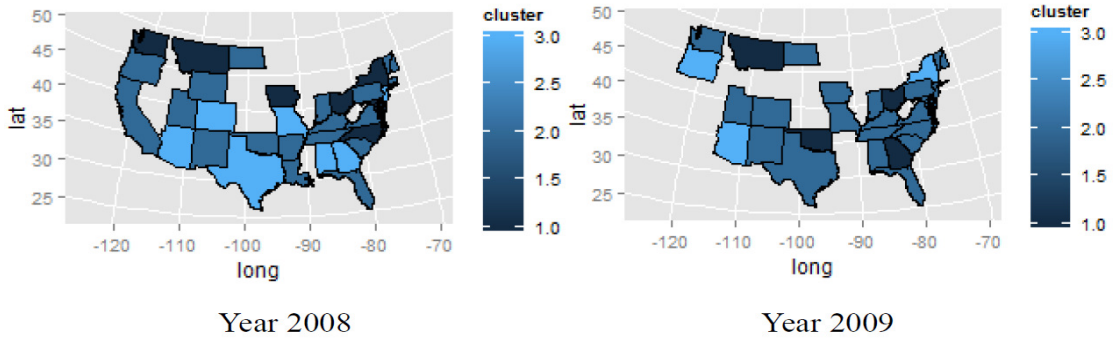
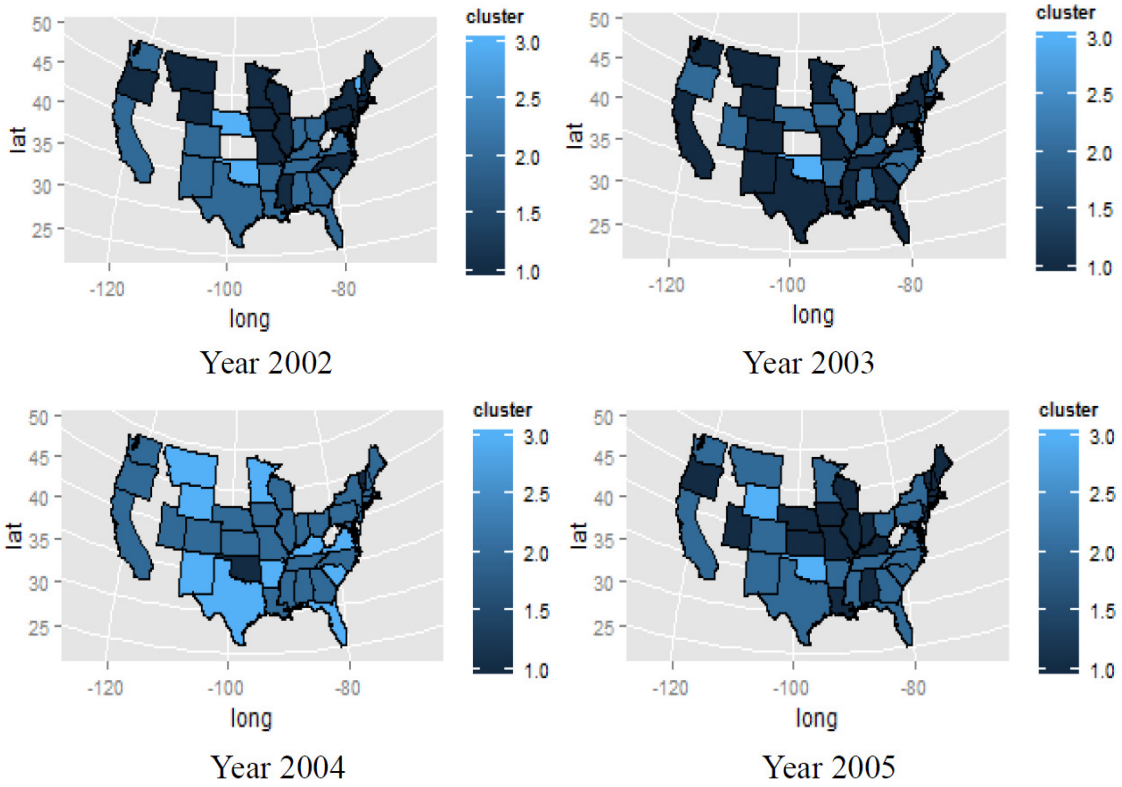


Figure 8. Clustering maps for public schools with out-of-state tuition using quantiles in every single year

From Figure 8 we can see that for the northeast coast, there is a sharp rise in year 2003 and year 2009 separately. This maybe because the Iraq War starts from March, 2003 and costs the government too much money, resulting in a budget cut for universities. Meanwhile, the financial crisis starts from 2008, causing a shortage of funds in universities. So tuition growth rates become higher. The similar situation also occurs in southern states, where the tuition growth rates are very high after year 2003. By contrast, tuition growth rates almost remain steady in west coast.

(3) Maps for private schools



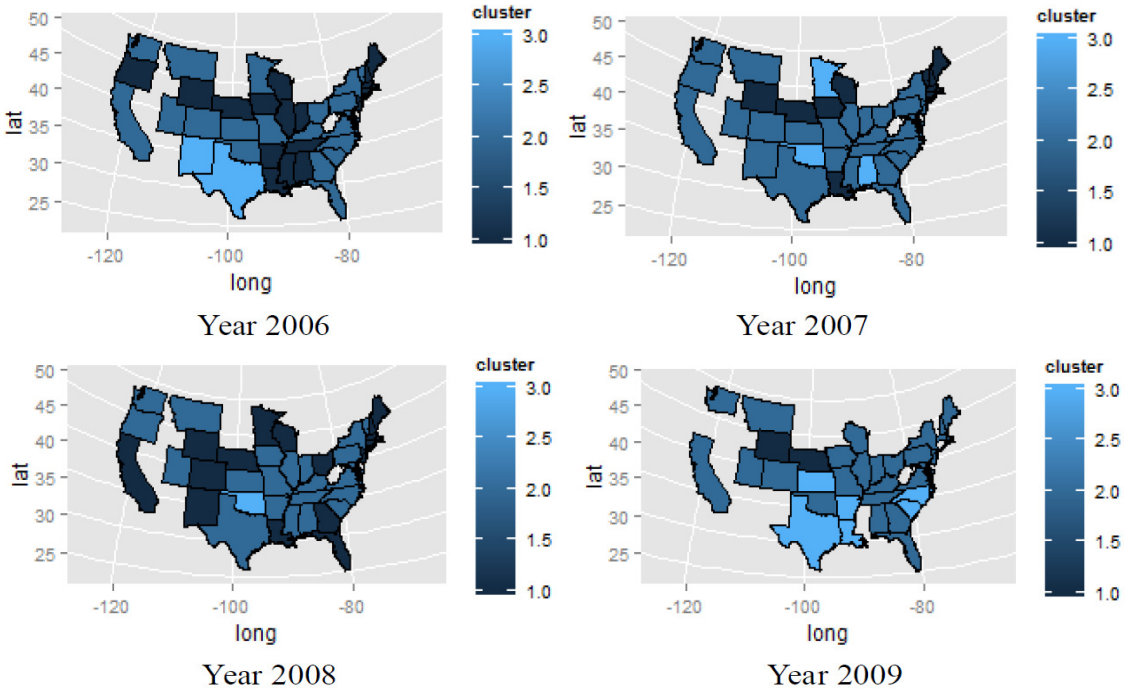


Figure 9. Clustering maps for private schools using quantiles in every single year

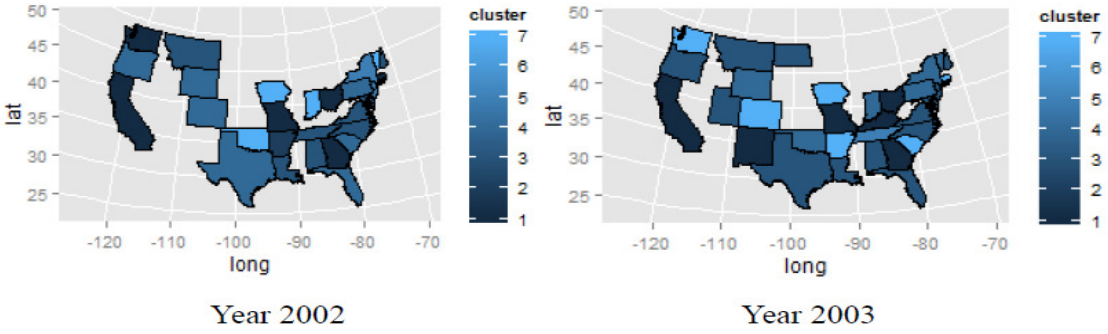
In Figure 9, it can be seen that from 2002 to 2009, the color of the middle area is first light, then it becomes darker and darker year by year, which indicates the tuition rates increase more and more slowly.

4.1.3 Using All Tuition Growth Rates

(1) Choose the number of clusters

We repeat same procedures as above, and we finally choose to make 7 clusters for both public schools with in-state tuition and public schools with-out-of-state tuition. And we do not do clustering analysis for private schools. To save place, this paper just shows maps for public schools with in-state tuition.

(2) Maps for public schools with in-state tuition



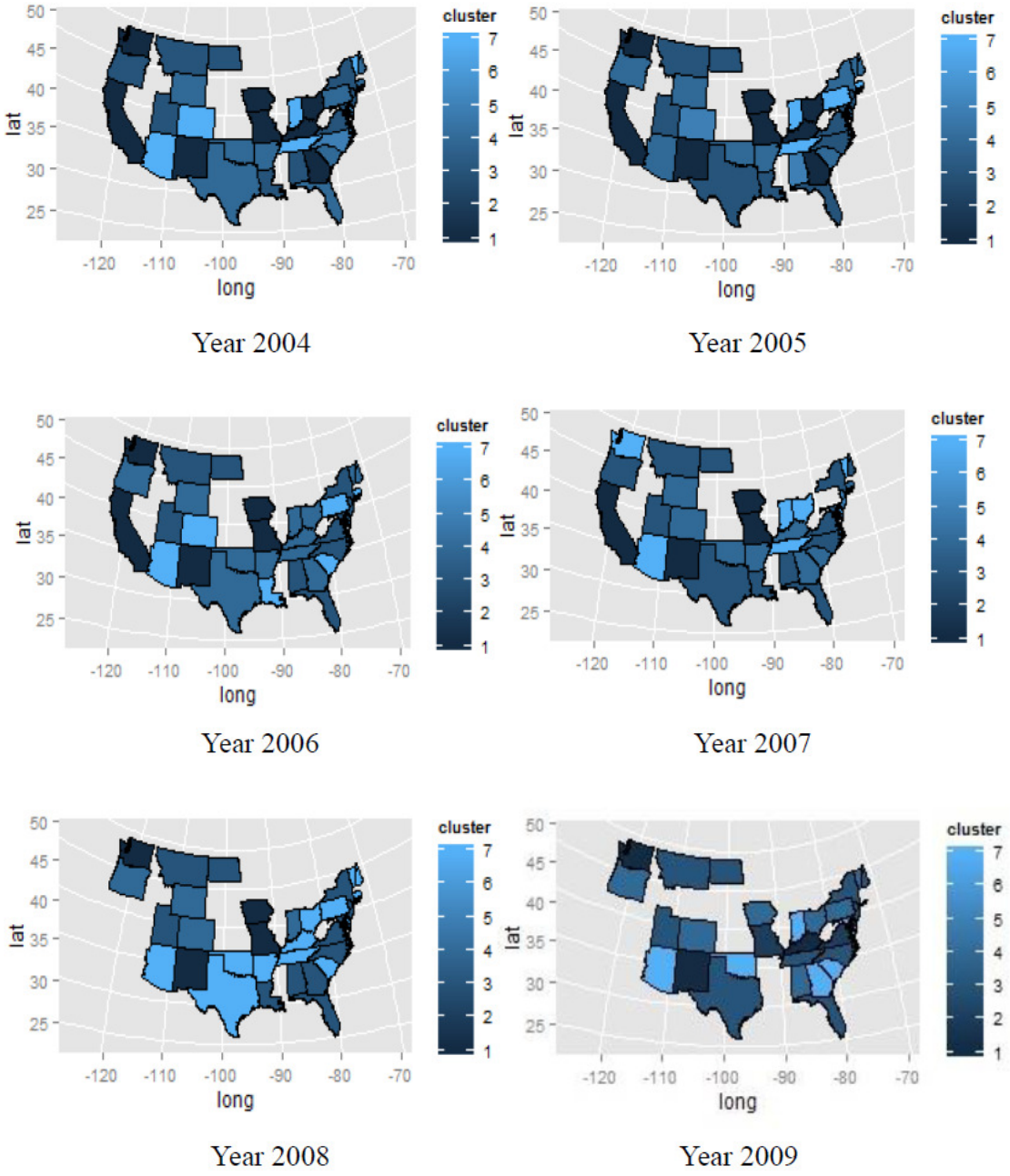


Figure 10. Clustering maps for public schools with in-state tuition using all tuition growth rates

4.1.4 Using Raw Tuition Fees

(1) Choose the number of clusters

The same procedures are repeated as above, 5 clusters are finally chosen for both public schools with in-state tuition and public schools with-out-of-state tuition using all raw tuition fees in the dataset. And we do not do clustering analysis for private schools. To save place, this paper just shows maps for public schools with in-state tuition.

(2) Maps for public schools with in-state tuition

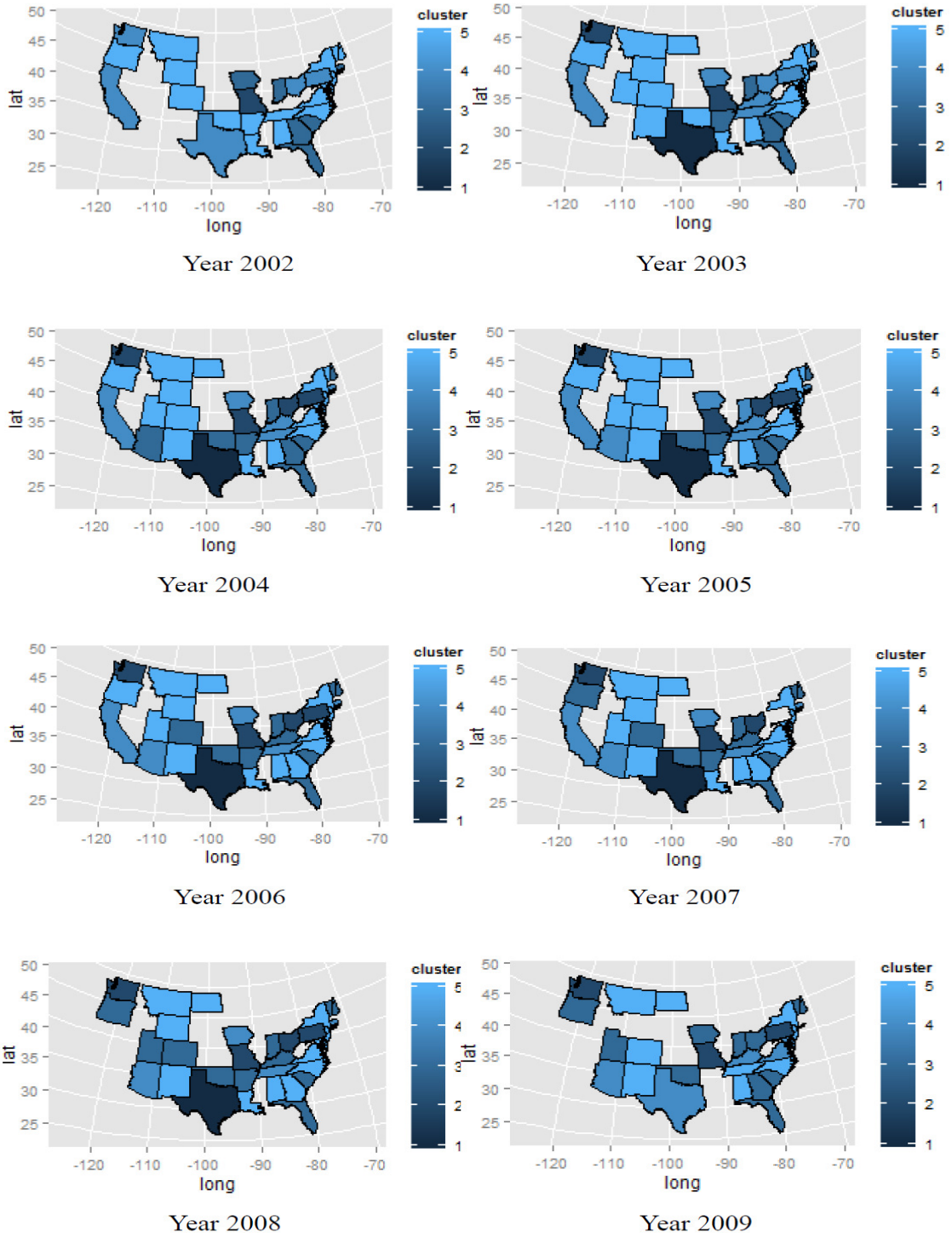


Figure 11. Clustering maps for public schools with in-state tuition using raw tuition fee

4.2 Regression Models

To make a more comprehensive analysis, we first winsorizes outliers because there are many unreasonable outliers in various variables, shown in Figure 11. Then we create dummy variables for academic year and use state and clusters as control variables. Also, we discover a collinear phenomenon existed between "employee to student ratio" and "faculty to student ratio". In addition, our regression model also uses some other variables including total revenue, expenditures, restricted revenue, states, academic years and so on. This paper uses linear regression, regression tree, decision tree and random effect model to study the rising tuition rates in the United States.

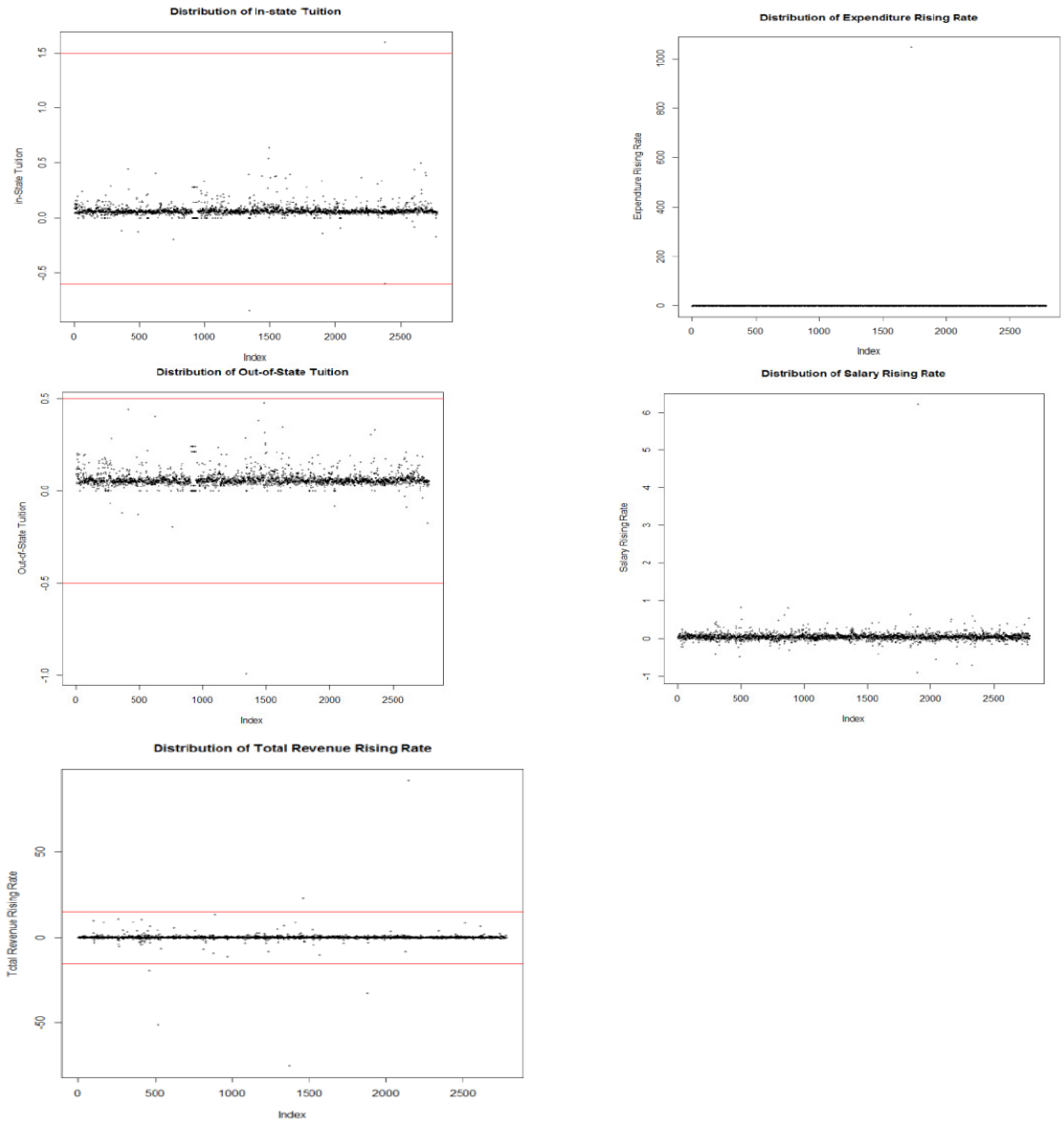


Figure 12. Data distributions and outliers for various variables

(1) Linear regression results

All models' R-squared is relatively small, which means this model does not fit the dataset very well. In addition, the predicted results using this linear regression model with the test dataset just capture a part of the trend about the tuition growth rates. The detailed regression results are shown in the following Tables.

Table 1. Regression results for all schools

tui03rr	Coef.	Std. Err.	t	P>t
eandgrr	0.0308656	0.0111142	2.78	0.006
totrevrr	-0.0023819	0.0014012	-1.7	0.09
empsturatiorr	0.007625	0.006951	1.1	0.273
salaryrr	-0.0145994	0.0105246	-1.39	0.166
resrr	-0.00053	0.0037814	-0.14	0.887

Table 2. Regression results for private schools

tui03rr	Coef.	Std. Err.	t	P>t
eandgrr	0.1158839	0.036665	3.16	0.002
totrevrr	0.0406108	0.0309585	1.31	0.193
empsturatiorr	0.0033604	0.0191222	0.18	0.861
salaryrr	-0.0418679	0.0269153	-1.56	0.124
resrr	-0.0052576	0.0187523	-0.28	0.78

Table 3. Regression results for public schools with out-of-state tuition

tui03rr	Coef.	Std. Err.	t	P>t
eandgrr	0.0167445	0.0094147	1.78	0.076
totrevrr	-0.0001127	0.0013266	-0.08	0.932
empsturatiorr	0.0111114	0.0071877	1.55	0.123
salaryrr	0.0039875	0.0099386	0.4	0.689
resrr	-0.0005065	0.003864	-0.13	0.896

Table 4. Regression results for public schools with in-state tuition

tui02rr	Coef.	Std. Err.	t	P>t
eandgrr	0.0295648	0.0581513	0.51	0.611
totrevrr	0.0289294	0.0405905	0.71	0.476
empsturationrr	-0.0033205	0.0301807	-0.11	0.912
salaryrr	-0.0508765	0.0166354	-3.06	0.002
resrr	-0.0205851	0.036119	-0.57	0.569

(2) Regression tree results
The regression trees results are shown as follows:
Variable importance for private schools:

```
  as.factor(state)                    eandgrr                empsturatio
        0.14752873                 0.13013811                 0.12212163
          salaryrr as.factor(academicyear)                    totrevrr
        0.10629947                 0.07977347                 0.04966225
              resrr
        0.03249592
```

Variable importance for public schools with in-state tuition:

```
 as.factor(state)                    salaryrr as.factor(academicyear)
        1.3318335                   0.9325881                  0.6918397
          eandgrr                    totrevrr                      resrr
        0.3097259                   0.2372871                  0.1698552
       empsturatio
        0.1635910
```

Variable importance for public schools with out-of-state tuition:

```
  as.factor(state) as.factor(academicyear)                empsturatio
        0.75903733                 0.35437101                 0.28087483
           eandgrr                    totrevrr                    salaryrr
        0.22667987                 0.11820407                 0.10002360
              resrr
        0.08607965
```

(3) Decision tree and random effect model

To account for the random effects in the analysis of tuition growth rates, we also propose a decision tree and random effect model in this paper. It is shown as follows:

$$y_i = f(\mathrm{X}_1,\mathrm{L}\,,\mathrm{X}_p) + \mathrm{Z}_i b_i + \varepsilon, \qquad i=1,\mathrm{L}\,,N \tag{1}$$

Fixed effects are described using decision tree model $f(X_1,L,X_p)$ and the random effect is considered using $Z_i b_i$. Though results of this model are very similar to the results in the regression tree section, this model can combine the merits of a tree-based model and a random effect model, which means that this model is capable of analyzing dataset with irregular data, tolerating variable selection bias and reducing computational cost.

5. CONCLUSION AND DISCUSSION

It has been noticed that the tuition growth rates have a sharp increase from 2003-2004 for public schools with in-state tuition, a sharp increase from 2004-2005 for public schools with out-of-state tuition, and a sharp increase from 2008-2009 for private schools. We deduce that these phenomenon are correlated with 2003 Iraq War, and 2008 Financial Crisis. From the clustering analysis, we find out that for private schools the tuition growth rates increase year by year for nearly all states. And for public schools, tuition growth rates of public schools on the west coast grow more slowly than that of public schools on the east coast. From the regression models, we can conclude that for all types of schools, both expenditures and total revenue are significant for the tuition growth rate. In addition, the tuition growth rate is positively related with the expenditure while negatively related with the total revenue.

REFERENCES

[1] Ehrenberg, Ronald G, (2012) "American higher education in transition", The Journal of Economic Perspectives, pp193-216.

[2] Bell, Daniel, (1976) "The coming of the post-industrial society", The Educational Forum, Vol. 40, No. 4. Taylor & Francis Group.

[3] Garrett, Thomas A, and William Poole, (2006) "Stop paying more for less: ways to boost productivity in higher education", The Regional Economist, Vol. 4, No. 9.

[4] Chiodo, Abbigail, and Michael T. Owyang, (2003) "Financial aid and college choice", Economic Synopses, 2003-08-03.

[5] Chang, W. (2012), R graphics cookbook, "O'Reilly Media, Inc.", ISBN: 978-1-449-31695-2.

[6] U.S. Department of Education, Institute of Education Sciences, National Center for Education Statistics. Data Source Website: https://nces.ed.gov/ipeds/deltacostproject.

[7] Heller, Donald E, (1996) "Tuition Prices, Financial Aid, and Access to Public Higher Education: A State-Level Analysis".

[8] Baird, Katherine, (2006) "Access to College: The Role of Tuition, Financial Aid, Scholastic Preparation and College Supply in Public College Enrollments", Journal of Student Financial, Vol. 36, No. 3, pp16-38.

[9] Betts, Julian R., and Laurel L. McFarland, (1995) "Safe port in a storm: The impact of labor market conditions on community college enrollments", Journal of Human Resources, pp741-765.

[10] Jackson, Gregory A., and George B. Weathersby, (1975) "Individual demand for higher education: A review and analysis of recent empirical studies", The Journal of Higher Education, pp623-652.

[11] Campbell, Robert, and Barry N. Siegel, (1967) "The demand for higher education in the United States, 1919-1964", The American Economic Review, pp482-494.

[12] Sewell, William H, (1971) "Inequality of opportunity for higher education", American Sociological Review, Vol. 36, No. 5, pp793-809.

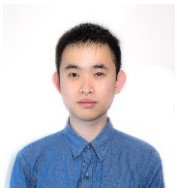
[13] Heckman, James J., Lance Lochner, and Christopher Taber, (1998) "General equilibrium treatment effects: A study of tuition policy", No. w6426, National Bureau of Economic Research.

[14] Becker, Gary S, (2009) "Human capital: A theoretical and empirical analysis, with special reference to education", University of Chicago Press.

[15] Heckman, James J, (1997) "Instrumental variables: A study of implicit behavioral assumptions in one widely used estimator", Journal of Human Resources, Vol. 32, No. 3.

AUTHOR

Long Cheng is currently working as the COO and research scientist at Kiwii Power Technology Co., Ltd, Troy, NY, USA. Before that, he worked as a data scientist at Rang Technologies Inc, Piscataway, NJ, USA. He received his Master's Degrees in both Electrical Engineering and Applied Mathematics from Rensselaer Polytechnic Institute, Troy, NY, USA in May 2015 and his B.S. in Electrical Engineering and Automation from Tianjin University, Tianjin, China in July 2013. His research interests include machine learning, data mining and smart grids.



Chenyu You is currently pursuing his B.S. in Electrical Engineering with a minor in Mathematics from Rensselaer Polytechnic Institute, Troy, NY, USA. His research interests are machine learning, data mining, statistical signal processing and mathematical modelling

PERFORMANCE EVALUATION OF TRAJECTORY QUERIES ON MULTIPROCESSOR AND CLUSTER

Christine Niyizamwiyitira and Lars Lundberg

Department of Computer Science and Engineering,
Blekinge Institute of Technology
SE-37179 Karlskrona, Sweden
cnw@bth.se, llu@bth.se

ABSTRACT

In this study, we evaluate the performance of trajectory queries that are handled by Cassandra, MongoDB, and PostgreSQL. The evaluation is conducted on a multiprocessor and a cluster. Telecommunication companies collect a lot of data from their mobile users. These data must be analysed in order to support business decisions, such as infrastructure planning. The optimal choice of hardware platform and database can be different from a query to another. We use data collected from Telenor Sverige, a telecommunication company that operates in Sweden. These data are collected every five minutes for an entire week in a medium sized city. The execution time results show that Cassandra performs much better than MongoDB and PostgreSQL for queries that do not have spatial features. Statio's Cassandra Lucene index incorporates a geospatial index into Cassandra, thus making Cassandra to perform similarly as MongoDB to handle spatial queries. In four use cases, namely, distance query, k-nearest neigbhor query, range query, and region query, Cassandra performs much better than MongoDB and PostgreSQL for two cases, namely range query and region query. The scalability is also good for these two use cases.

KEYWORDS

Databases evaluation, Trajectory queries, Multiprocessor and cluster, NoSQL database, Cassandra, MongoDB, PostgreSQL.

1. INTRODUCTION

Large scale organisations continuously generate data at very high speeds. These data are often complex and heterogeneous, and data analysis is a high priority. The challenges include what technology in terms of software and hardware to use in order to handle data efficiently. The analysis is needed in different fields such as transportation optimization, different business analytics for telecommunication companies that seek to know common patterns from their mobile users.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 145–163, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60612

The analysis comprises querying some points of interests in big data sets. Querying big data can be time consuming and expensive without the right software and hardware. In this paper, various databases were proposed to analyse such data. However, there is no single database that fits all queries. The same holds for hardware infrastructure there is no single hardware platform that fits all databases. We consider a case of trajectory data of a telecommunication company where analysing large data volumes trajectory data of mobile users becomes very important. We evaluate the performance of trajectory queries in order to contribute to business decision support systems.

Trajectory data represents information that describes the localization of the user in time and space. In the context of this paper, a telecommunication company wants to optimize the use of cell antennas and localize different points of interests in order to expand its business. In order to successfully process trajectory data, it requires a proper choice of databases and hardware that efficiently respond to different queries.

We use trajectory data that are collected from Telenor Sverige (a telecommunication company that operates in Sweden). Mobile users' position is tracked every five minutes for the entire week (Monday to Sunday) from a medium size city. We are interested to know how mobile users move around the city during the hour, day, and the week. This will give insights about typical behavior in certain area at certain time. We expect periodic movement in some areas, e.g., at the location of stores, restaurants' location during lunch time.

Without loss of generality, we define queries that return points of interests such as nearest cell location from a certain position of a mobile user. The contribution of this study is to solve business complex problem that is,

- Define queries that optimize the point of interests, e.g., nearest point of interest, the most visited place at a certain time, and more.
- Choice of database technology to use for different types of query.
- Choice of hardware infrastructure to use for each of the databases.

This data is modelled as spatio-temporal data where at a given time t a mobile user is located at a position (x, y). The location of a mobile user is a triples (x, y, t) such that user's position is represented as a spatial-temporal point p_i with $p_i = (x_i, y_i, t_i)$.

By optimizing points of interests, different types of queries are proposed. They differ in terms of what are their input and output:

- *Distance query* which finds points of interests that are located in equal or less than a distance, e.g., one kilometer from a certain position of a mobile user.

- *K-Nearest neighbor* query that finds K nearest points of interests from a certain position of a mobile user.

- *Range query* that finds points of interests within a space range from a certain position of a mobile user.

- *Region query* that finds the region that a mobile user frequently passes through at certain time throughout the week.

The performance of different queries is evaluated on three open sources databases; *Cassandra, MongoDB*, and *PostgreSQL*. We choose to use open source databases for the sake of allowing the study reproducibility. The hardware configuration is done on a single node, and on multiple nodes (distributed) in a cluster. The execution time of each of the queries at each database on different hardware infrastructure is measured. Since the company knows the locations that are the most, or the least visited during a certain time, in order to avoid overloading and underloading at such locations, antenna planning will be updated accordingly. For business expansion, a busy location during lunch time is for e.g., convenient for putting up a restaurant. Moreover, the performance measurement shows which database is better for which specific query on which hardware infrastructure, thus contributing to business support systems.

The rest of the paper is organized as follows; Section 2 defines the concepts, Section 3 summarizes the related work, Section 4 describes the configuration and gives the databases overview, Section 5 presents results and discussions, finally Section 6 draws conclusions.

2. TRAJECTORY DATA

2.1 Definition of Trajectory

Trajectory is a function from a temporal domain to a range of spatial values, i.e., it has a beginning time and an ending time during which a space has been travelled (see Equation 1)[1].

$$[t_{begin}\ t_{end}] \rightarrow space \tag{1}$$

A complete trajectory is characterized by a list of triples $p = (x, y, t)$, thus a trajectory is defined as a sequence of positions T_{pos}.

$$T_{pos} = \{p_1, p_2, \dots, p_n\} \tag{2}$$

where $p_i = (x_i, y_i, t_i)$ represents a spatio-temporal point, Figure 1 shows such trajectory.

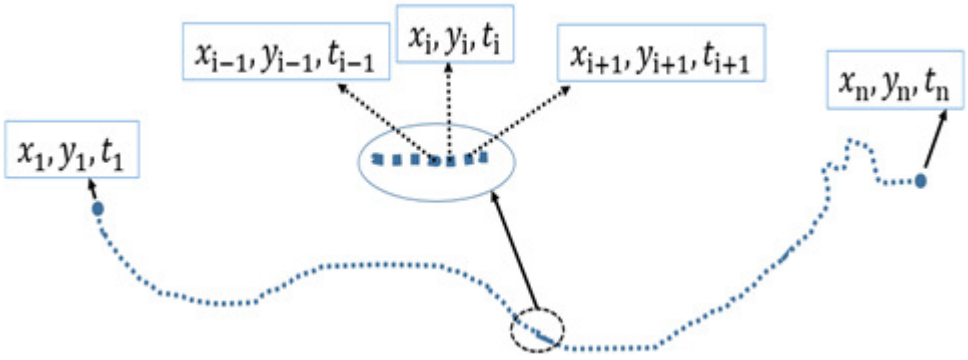


Figure 1. Mobile user's trajectory as a sequence of triples

In this study, the trajectory data has space extent that is represented by latitude and longitude. With x represents latitude and y represents longitude, and the time that is represented by t; i.e., a mobile user is located at position (x, y) at time t

2.2. Definition of Trajectory Queries

Trajectories queries are historical spatio-temporal data which are the foundation of many applications such as traffic analysis, mobile user's behavior, and many others [2], [3]. Trajectories queries make analytics possible, e.g., mobile users' positions at a certain time. In the context of location optimization, common trajectory queries that we consider in this study are following; Distance query, Nearest neighbor query, Range query, and Region query.

Figure 2 describes query types, where C_i represents different cell-city names, each C_i is represented by (x_i, y_i) where x_i is latitude and y_i is longitude. Distance query returns cell-cities that are located at a distance from C_1, e.g., within distance L from the position of C_1. The query returns C_2, C_3, C_4, C_7.

At a given fixed time or time interval, we are retrieving two cell-cities that are the most close to city C_1, that is K-NN query with $k = 2$. Given a space range$[B, E]$, range query returns the cell-cities that belong to that space range.

Region query returns the cell-city that a given user frequently visits. e.g., user *Bob* passes mostly through cell-city C_8 (see Figure 2).

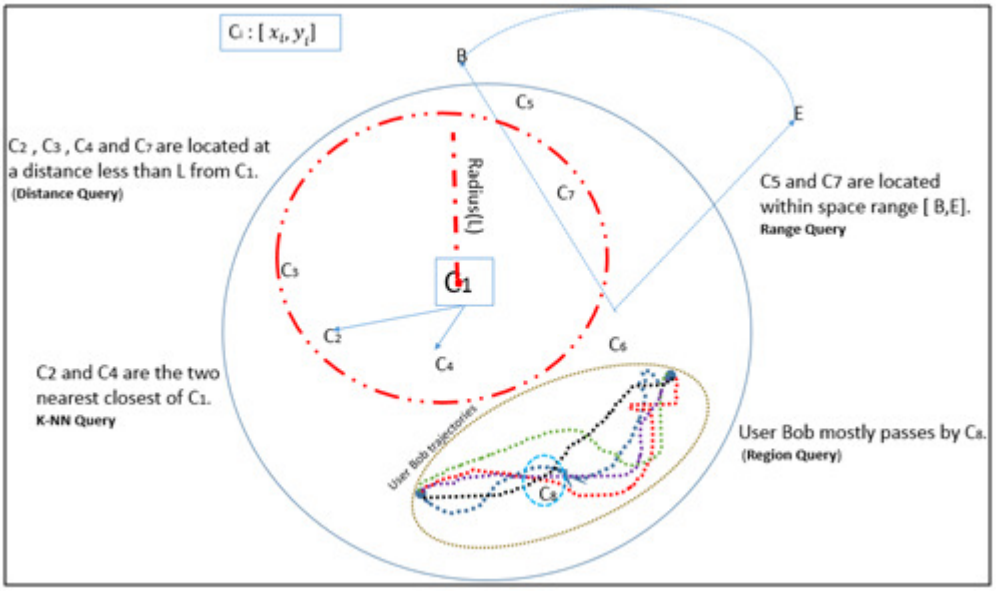


Figure 2. Visualization for Query Types

2.2.1. Distance Query

Definition: Distance query returns all point of interest (e.g., gas stations) whose distance (according to a distance metric) from a given position that is less than a threshold [2], [3]. Figure 3 shows inputs to distance query.

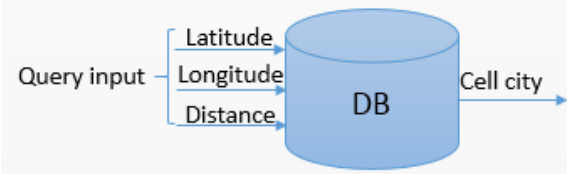


Figure 1. Distance Query

Example: find cells that are located in less than 1km from a certain mobile user's position.
In terms of latitude and longitude coordinates, a query that covers the circle of 10 km radius from a user position at $(x_p, y_p) = (1.3963, -0.6981)$ is expressed for different databases as follows;

- In Cassandra

$$SELECT \text{ cell\_city } FROM\ mobility\ WHERE\ expr(info\_index, '\{filter: \{type: "boolean", must:$$

$$[\{type: "geo\_distance", field: \text{place}, latitude: 1.3963, longitude: -0.6981, max\_distance: "10km"\}\ ]\ \}\}')$$

- In MongoDB

$$db.Mobility.find\ (\ \{\ location: \{\ \$near : [-0.6981, 1.3963], \$maxDistance: 10\ \}\ \}, \{Cell\_CITY: 1\})$$

- In PostgreSQL

$$SELECT\ cell\_city\ FROM\ Mobility\ WHERE$$
$$\arccos\left(\sin(x_p) * \sin(x) + \cos(x_p)\right) * \cos(x) * \cos(y - (y_p))) * R <= 10\ ; \quad (3)$$

With R is the radius of earth, $R = 6371$ km

In order to index latitude and longitude columns so that the database understand the query, the circle of radius 10 km is bounded by a minimum and a maximum coordinates, let's say $p_{min} = (x_{min}, y_{min})$ and $p_{max} = (x_{max}, y_{max})$, then query (3) becomes as follows;

$$SELECT\ Cell\_city\ FROM\ mobility\ WHERE\ (x => x_{min}\ AND\ x <= x_{max})\ AND$$
$$(y >= y_{min}\ AND\ y <= y_{max})\ AND$$
$$arccos(sin(x_p) * sin(x) + cos(x_p) * cos(x) * cos(y - (y_p))) <= r\ ;$$

With r is the angular radius of the query circle,
$r = distance/earth\ radius$
$x_{min} = x - r$
$x_{max} = x + r$
$y_{min} = y - \Delta y$
$y_{max} = y + \Delta y$
$\Delta y = \arccos(\ (\cos(r) - \sin(x_T) * \sin(x)\ )\ /\ (\cos(x_T) * \cos(x)\ )\ )$

With $x_T = \arcsin(\sin(x)/\cos(r))$. More on positions' angles calculation is found in [4].

2.2.2. k Nearest Neighbor Query

Definition: k-Nearest Neighbor (KNN) Query returns *k* points of interest which are the closest to a given position (x, y) [5], [6]; and *k* results are ordered by proximity. KNN can be bounded with a distance, in that case, KNN behaves like a distance query, if k is not indicated.
Figure 4 shows inputs to kNN query, where kNN is bounded within a distance.

Figure 4. kNN query

Example: find five nearest cells from a mobile user's position. A typical query that select 5 nearest cells within 10 km is as follows;

- In Cassandra

$$SELECT \text{ cell\_city } FROM\ Mobility\ WHERE\ expr(info\_index, '\{\ filter: \{type: "boolean", must: [\{type: "geo\_distance", field: "place", latitude: 1.3963, longitude: -0.6981, max\_distance: "10km"\}\ ]\ \}\}')\ \ LIMIT\ 5$$

- In MongoDB

$$db.Mobility.find(\ \{location: \{\$near: [-0.6981, 1.3963], \$maxDistance: 0.10\ \}\ \}, \{\text{cell\_city: } 1\}).limit(5)$$

- In PostgreSQL

$$SELECT\ \ Cell\_city\ FROM\ mobility\ WHERE\ (x\ =>\ x_{min}\ AND\ x\ \leq\ x_{max})\ AND\ (y\ >=\ y_{min}\ AND\ y\ < = y_{max})\ AND arccos\left(sin(x_p) * sin(x) + cos(x_p) * cos(x) * cos\left(y - (y_p)\right)\right) \leq r\ \ ORDER\ BY\ x\ ASC, y\ ASC\ LIMIT\ 5$$

2.2.3. Range Query

Definition: Range query returns all point of interest (e.g., gas stations) that are located within a certain space shape (polygon) [2].

Figure 5 shows inputs to range query to find cells that belong to a polygon.

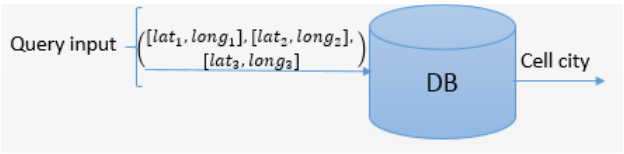


Figure 2. Range query

Example: find cells within a space range that is indicated by a polygon from a mobile user position [longitude, Latitude] coordinates.

A typical query that select cells that are located within a geographical bounding box (polygon shape), e. g, a triangle from a mobile user position coordinates: [12.300398, 57.569256] within ([11.300398, 56.569256], [12.300398, 58.569256]) is as follows;

- In Cassandra

$$SELECT \text{ cell\_city } FROM\ Mobility\ WHERE\ expr(info\_index, '\{ filter: \{type: "boolean", must: [\{type: "geo\_bbox", field: "place", min\_latitude: 11.300398, max\_latitude: 12.300398 min\_longitude: 56.569256, max\_longitude: 58.569256 \}] \}\}')$$

- In mongoDB

$$db.Mobility.find(\{ location: \{ \$geoWithin: \{ \$polygon: [[12.300398, 57.569256], [11.300398, 56.569256], [12.300398, 58.569256]] \}\}\}, \{Cell\_City\})$$

- PostgreSQL

The following is a typical query range query between two points $p_{min} = (x_{min}, y_{min})$ and $p_{max} = (x_{max}, y_{max})$ with x is latitude and y is longitude.

Example: $p_{min} = (1.2393, -1.8184)$ and $p_{max} = (1.5532, 0.4221)$.

$$SELECT\ \ cell\_city\ FROM\ mobility\ \ WHERE\ (x\ =>\ 1.2393\ AND\ \ x\ \leq\ 1.5532)\ AND\ (y\ >=\ -1.8184\ AND\ \ y\ <=\ 0.4221)$$

2.2.4. Region Query

Generally, trajectories of mobile users are independent each other, however, they contain common behavior traits such as passing through a region at a certain regular period, e.g., passing through the shopping center during lunch time.

Definition: identify the region which is more likely to be passed by a given user at a certain time based on the many other relevant regions to that user [2]. In the context of this study, the knowledge about region reveals which cell city that many users mostly pass by, this cell city might have some point of interests such as stores, high way junction.

Figure 6 shows inputs to region query to find cell city that is the most visited at certain time.

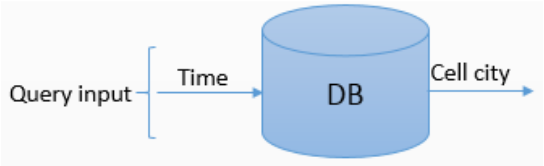


Figure 3. Region query

Example: find the cell city that is frequently passed by the same mobile users during a certain time every day for the entire week. A typical query that returns the cell city that is the most visited during interval [12: 10: 00,13: 10: 00] is as follows;

- In Cassandra

$$SELECT\ DISTINCT\ \text{cell\_city}\ FROM\ \ Mobility\ \ where\ Time = '12{:}10{:}00'\ and\ Time < = '13{:}10{:}00'\ GROUP\ by\ \text{cell\_city}\ ORDER\ BY\ "count"\ DESC$$

- In MongoDB

$$db.Mobility.find(\ \{Time{:}\ \{\$gt{:}\ '12{:}10{:}00', \$lt{:}\ '13{:}10{:}00'\}\},\ \{\text{cell\_city}{:}\ 1\}\ ).distinct().count()$$

- In PostgreSQL

$$SELECT\ \text{cell\_city}, count(*)\ \ from\ mobility\ WHERE\ t \geq 12{:}10{:}00\ and\ t \leq 13{:}10{:}00\ \ GROUP\ by\ \text{cell\_city}\ ORDER\ BY\ "count"\ DESC$$

3. RELATED WORK

In [7], authors propose an approach and implementation of spatio-temporal database systems. This approach treats time-changing geometries, whether they change in discrete or continuous steps. The same can be used to tackle spatio-temporal data in the other databases. We rather evaluate trajectory queries on existing general purpose databases notably Cassandra, PostgreSQL, and MongoDB. In [8], author describes requirements for database that support location based-service for spatio-temporal data. A list of ten representative queries for stationary and moving reference objects are proposed. Some of those queries that are related to this study are given in section two.

In [9], Dieter studied trajectory moving point object, he explained three scenarios, namely constrained movement, unconstraint movement and movement in networks. Different techniques to index and to query these scenarios define their respective processing performance. Author modelled the trajectory as triples (x,y,t), we use the same model in this study.

In [10], authors introduced querying moving objects (trajectory) in SECONDO, the latter is a DBMS prototyping environment particularly geared for extension by algebra modules for nonstandard applications. The querying is done using SQL-like language. In our study, we are querying moving object using SQL and Not Only SQL (NoSQL) querying languages on top of different databases. Continuously, authors provide a benchmark on range queries and nearest

neighbor queries on SECONDO DBMS for moving data object in Berlin. The moving object data was generated using computer simulation based on the map of Berlin [11]. This benchmark could be extended to other queries such as region queries, distance queries, and so on. In our study, we apply these queries on real world trajectory data, i.e., mobile users' trajectory from Telenor Sverige.

In [5], authors introduced a new type of query Reverse Nearest Neighbor (RNN) which is the opposite to Nearest Neighbor (NN). RNN can be useful in applications where moving objects agree to provide some kind of service to each other, whenever a service is need it is requested from the nearest neighbor. An object knows objects that it will serve in the future using RNN. RNN and NN are relatively represented by distance query in our study. In [12], authors studied aggregate query language over GIS and no-spatial data stored in a data warehouse. In [13], authors studied k-nearest search algorithm for historical moving object trajectories, this k-nearest neighbor is one of the queries that is considered in our study.

In [14], authors presents techniques for indexing and querying moving object trajectories. This data is represented as three dimension (3D) space, where two dimensions correspond to space and one dimension corresponds to time. We also represent our data in 3D as (x,y,t), with x,y represents space whereas t represents time.

Query processing on multiprocessor has been studied in [15], authors implemented an emulator of parallel DBMS that uses cluster with multiprocessor. This study is different from ours in a sense that we evaluate query processing on real physical hardware with existing general purpose databases. Query processing on FPGA and GPU on spatial-temporal data was studied in [16]. Authors present a FPGA and GPU implementation that process complex queries in parallel, the study did not investigate the performance on various existing databases, the distributed environment was not also considered, whereas, in our study we investigate query processing on various databases on top of different computational platforms including cluster. In [17], authors conducted a survey on mining massive-scale spatio-temporal trajectory data based on parallel computing platforms such as GPU, MapReduce and FPGA, again existing general purpose databases were not evaluated. Authors presented a hardware implementation for converting geohash codes to and from longitude/latitude pairs for Spatio-temporal data [18], the study shows that longitude and latitude coordinates are the key points for modelling spatio-temporal data. In our paper, we also use these coordinates for location based querying.

4. DATABASE OVERVIEW AND CONFIGURATION

The development of technology involves big data that is structured and unstructured. The presence of unstructured data stimulates the invention of new databases, since Relational Database Management Systems (RDBMS) that uses Structured Query Language (SQL) to deal with structured data only becomes unable to handle unstructured data. A new data model, Not Only SQL (NoSQL) was introduced to deal also with unstructured data [19]. Main features of NoSQL follow CAP theorem (Consistency, Availability, and Partition tolerance). The core idea of CAP is that a distributed system cannot meet the three distinct needs simultaneously. According to data models, NoSQL can be relational, key value based, column based, and document based. In this study we choose three open source databases that have diverse features of SQL (PostgreSQL) and NoSQL (Cassandra and MongoDB).

Key value data model means that a value corresponds to a key, column based uses tables as the data model, the data is stored by column, each column is the index of the database, queries are applied to column, whereby each column is treated one by one. Document based database stores in JSON or XML format, each document (similar to a row in RDBMS) is indexed and it has a key.

4.1. Cassandra

Apache Cassandra is an open-source NoSQL column based database. It is a top level Apache project born at Facebook and built on Amazon's Dynamo and Google's BigTable. It is a distributed database for managing large amounts of structured data across many commodity servers, while providing highly available service and no single point of failure. In CAP, Cassandra has availability and partition tolerance (AP) with eventual consistency. Cassandra offers continuous availability, linear scale performance, operational simplicity and easy data distribution across multiple data centers and cloud availability zones. Cassandra has a masterless ring architecture [23]. Keyspace is similar to database in RDBMS, inside keyspace there are tables which are similar to tables in RDBMS, column and rows are similar to those of RDBMS' tables. The querying language is Cassandra Query Language (CQL) is almost similar to SQL in RDBMS [24].

4.2. MongoDB

MongoDB is an open-source NoSQL document database, MongoDB is written in C++. MongoDB has database, inside a database there are collections, these are like table in RDBMS, Inside a collection there are documents, these are like a tuple/row in RDBMS, and inside a document there are fields which are like column in RDBMS [21], [22]. MongoDB is consistent and partition tolerant.

4.3. PostgreSQL

PostgreSQL is an open source Object RDBMS that has two features according to CAP theorem, those are availability, i.e., each user can always read and write. PostgreSQL consists of consistency, i.e., all users have the same view of data. PostgreSQL organises data in column and rows [20, p. 3].

4.4. Single Node Installation

Two types of server are used,
1. Hardware type 1: Dell powerEdge R320
Operating system: Ubuntu 14.04.3 LTS x86_64
RAM memory: 23 GB RAM
Harddisk size: 279.4GB 0 disk
Processor (Intel(R) Xeon(R) CPU E5-2420 v2) has 12 cores, each core is hyperthreaded into 2 cores, this give 24 virtual cores. These servers are exclusive, i.e., they are only running our databases.

2. Hardware type2: Fujitsu RX600S5
Operating system: Ubuntu 13.04 LTS X86_64
The RAM memory: 1024 GB.

Processor (4x Xeon X7550) has 32 cores, each core is hyperthreaded into 2 cores, this give 64 virtual cores. At the time of experiment this server is running some other work, i.e., it is not exclusive to our databases only. This affect the execution time of our databases, however the trends such as variability between queries upon the databases are not affected. Standard deviation of the execution time keeps the same trends.

4.5. Multiple Nodes Installation

A cluster is made up of 4 nodes, each node is hardware type 1 and it has the same features as the other.

Cassandra partitions and replicates data across 4 nodes in the cluster (see Figure 7). Since we are using a small cluster of four nodes with all nodes belong to the same rack and same data center, the replication strategy is set to simple strategy with four replicas across the cluster. SimpleStrategy places the first replica on a node that is determined by the partitioner. A partitioner determines how data is distributed across the nodes in the cluster including replicas. We configured partitioner as a Murmur3Partitioner, the latter provides faster hashing and improved performance. And the snitch that informs Cassandra about the network topology is configured as the simple snitch [27]. All the nodes in the cluster are peers, with one of the nodes is configured as a seed, the latter bootstraps the gossip process for new nodes joining the cluster. Each node has the same copy of data as the other, in this study we use Cassandra 3.0.3. The replication factor equals to number of nodes. Since we have a single data center with no write activities because we only need to read the given data, we use a consistency level one, i.e., the closest replica node for the given row is contacted to respond the query.

Cassandra does not natively support spatial indexing but this can be extended via Stratio's Cassandra Lucene index. Stratio's Cassandra Lucene Index is a plugin for Apache Cassandra derived from Cassandra, it extends its index functionality to provide near real time search such as ElasticSearch or Solr, including full text search capabilities and free multivariable, geospatial and bitemporal search. We use Stratio's Cassandra Lucene Index 3.0.4 [28].

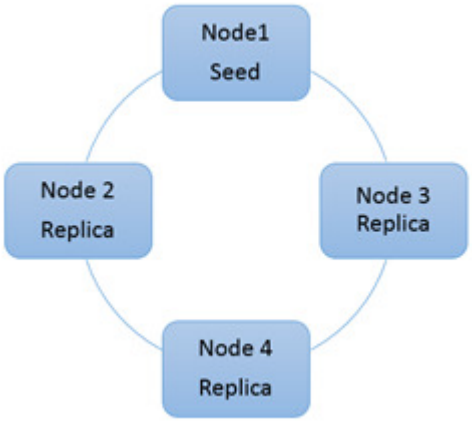


Figure 7. Cassandra Structure

MongoDB partitions data across nodes, i.e., MongoDB scales horizontally by dividing and distributing data over multiple servers that are called shards. Each shard is an independent database, and collectively, the shards make up a single logical database. Sharding reduces the

number of operations each shard handles. Each shard processes fewer operations as the cluster grows. As a result, a cluster can increase capacity and throughput horizontally (by adding nodes in the cluster). Sharding reduces the amount of data that each server needs to store. Each shard stores less data as the cluster grows [26]. Sharded cluster (contains shards, config servers and mongos instances). We use three shards, each on a node. In Figure 8, we see config servers that holds the metadata about the cluster such as the shard location of the data, they must be three servers. There is also Mongos server that serves as the routing service that process queries throughout the cluster. Mongos is installed on its own node, whereas config servers and shards are installed on the same nodes (1, 2, 3) as it shown in the Figure 8. Since we have three shards, each shard contains a third of the total data. Mongo will be eventually available if we replicate each shard on different nodes. In this study we install MongoDB 3.0.9. MongoDB has built in spatial query functions.

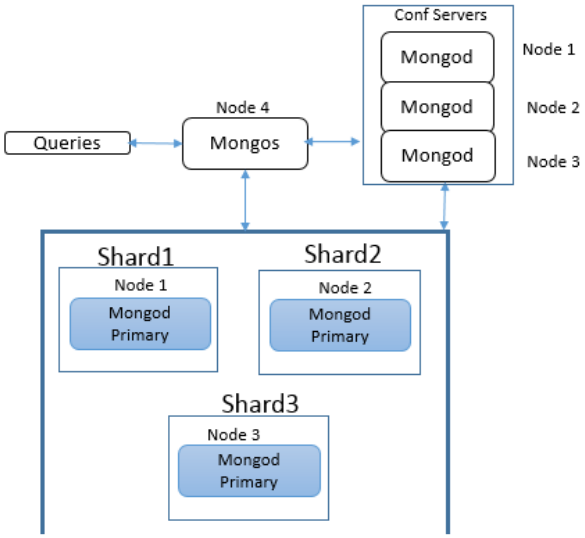


Figure 4. MongoDB Structure

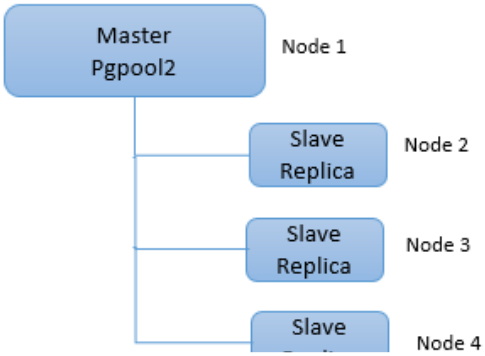


Figure 5. PostgreSQL Structure

PostgreSQL is installed on the cluster in master/ slave replication mode (see Figure 9). Nodes serve each other in pool using pgpool2 [25]. Pgpool 2 provides, load balancing and data redundancy. In order to keep available the data, each slave holds a copy of data and it is read-only, there are three slaves, thus three replicas. Whereas in order to keep the consistency of the, only the master can read and write. Master and pgpool 2 are installed on the same node. In this

study we install PostgreSQL version 9.3.11. PostgreSQL does not have explicit spatial query functions, thus, we have to use mathematical functions in order to query the database using geographical coordinates.

4.6. Data Description

The mobility or location update is generated when a handset is generating traffic either downloading or uploading. Mobility is captured in five minutes intervals and include all cells during those five minutes. Mobility is indicated by number of cells within timeframe and the distance between those cells.

The data we use in this paper is collected every five minutes for the entire week in a small medium size city. We have a collection of two millions five hundred ninety three thousands three hundred sixty records (2,593,360) for different users. Every record has eighteen attributes (18 x 2,593,360). Those attributes are; UserId, SiteId, weekday, Time, ProfileId, SegmentId, SourceGSM, SourceUMTS, SourceLTE, Easting, Northing, Latitude decimal, Longitude decimal, Cell municipality, Cell county, Cell city, Cell postcode, Cell address. This data is populated in Cassandra and PostgreSQL without any transformation. Whereas, in MongoDB, coordinates attributes (latitude and longitude) were combined into an array location attribute in order to be able to use built in spatial function in MongoDB.

5. RESULTS AND DISCUSSION

Figures 10, 11, 12, 13 show the execution time with respect to different number of nodes in the cluster, we present results using logarithmic scale. All the results are the average of ten runs of each query. More detailed data are given in appendix in Tables 1, 2, 3, 4, 5. Those tables show the execution time for different queries on Cassandra, MongoDB, and PostgreSQL databases on four, three, two, and one nodes for hardware type one and type two respectively.

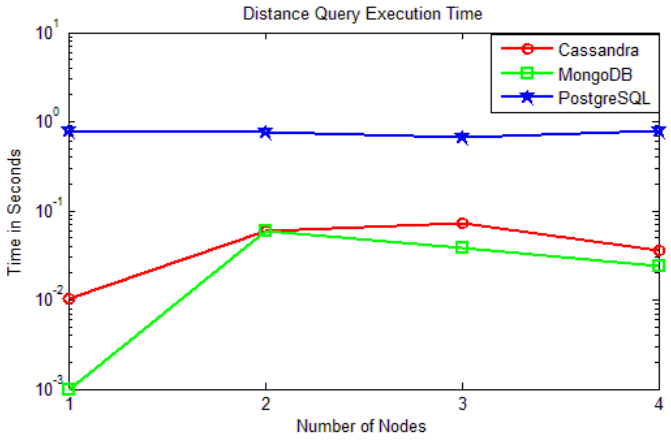


Figure 10. Distance Query Execution Time

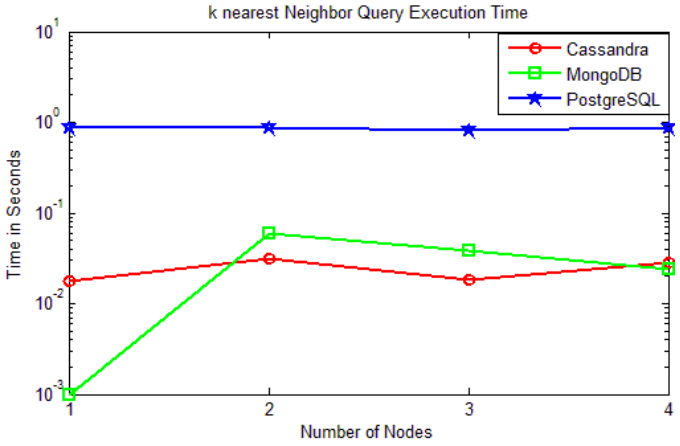


Figure 6. K nearest Neighbor Query Execution Time

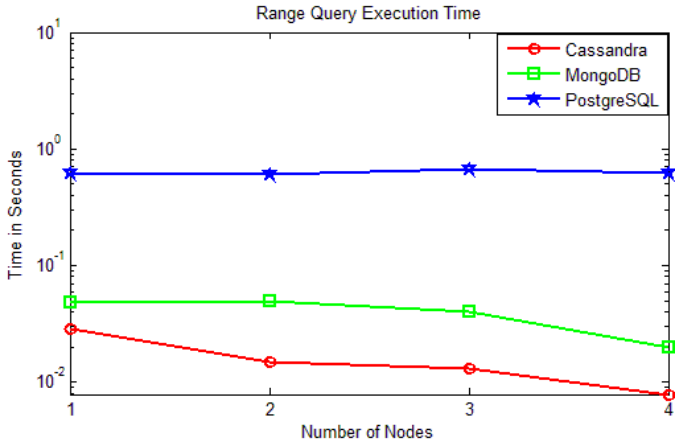


Figure 7. Range Query Execution Time

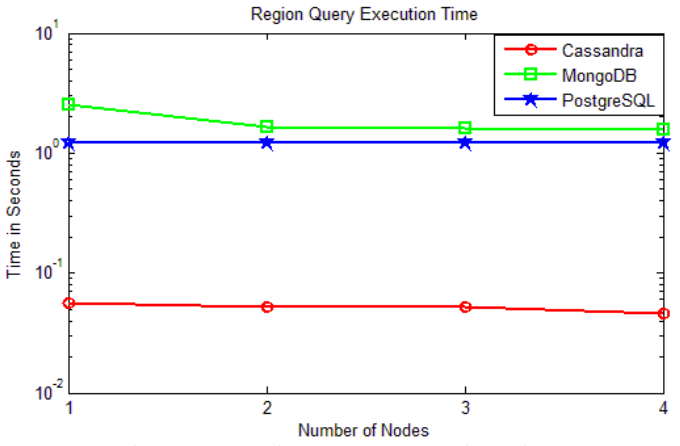


Figure 13. Region Query Execution Time

It is observed that Cassandra has the shortest execution time for range and region queries, particularly for region query. Region query has one input which is time, it does not involve spatial features or geographical shapes, e.g., sphere, near, within. It is clear that Cassandra outperforms much better than MongoDB and PostgreSQL for general purpose queries. E.g., Region query involves time only as input. For queries that contain geographical or specific spatial features, MongoDB seems to perform almost as Cassandra when the latter is indexed by Stratio Lucene Index (see Figure 10, 11, 12). In figure 13, MongoDB has longer execution than Cassandra and PostgreSQL for region query, this is caused by aggregation query process which seems to be slower in MongoDB.

In all queries, when we run a query for the first time, we observed that for Cassandra, it takes longer than the next runs. We have to mention that Lucene makes a huge use of caching, therefore the first query will be especially slow due to the cost of initializing caches [26]. Thus, we disregard the first run of each query when measuring the performance.

Whereas for MongoDB and PostgreSQL the same query on the same hardware, runs almost with relatively same execution time. Spatial queries have the longest execution time in PostgreSQL, the reason is that we have to use mathematical functions to represent geographical locations. This involves different steps of calculation, thus, making it longer (see Figure 10, 11, 12).

The scalability according to increase of number of nodes is significant for Cassandra and MongoDB for range query. The reason is that the range query involves a partition of the data according to range specification, hence the cache is relatively not overloaded. Whereas the scalability is not very noticed for the other queries which covers the whole data, thus consuming much cache which results in slowing the execution time. In terms of processing, PostgreSQL does not exploit the increase of number of nodes, since nodes are used for replication purposes in order to keep the database available. MongoDB distributes data across shards, in order to provide high availability, we need to replicate each shard on its own server, e.g., in our case we have three shards, in order to have a second copy of the whole data, we need three more servers, in total we need 6 servers for 2 copies. However, for Cassandra, since we have a full copy of data at each node, i.e., for 4 nodes cluster we have 4 copies of data. This feature makes Cassandra to be attractive than MongoDB in cases where a number of servers is a constraints. Furthermore if Mongos fails, the whole database fails, the same holds for PostgreSQL, if the master goes down, the whole database cannot operate anymore. Whereas, for Cassandra, if any node goes down, others keep working.

6. CONCLUSIONS

In this study, we evaluated the performance of trajectory queries on Cassandra, MongoDB, and PostgreSQL on Multiprocessor and cluster environment. The evaluation is conducted on data collected from a Telecommunication Company. We observed that Cassandra performs much better than MongoDB and PostgreSQL to handle queries that do not contain special geographical features such as sphere shape, near coordinates (example of region query that involves time as input). MongoDB has natively a built in function for spatial queries, this speeds up the query response time. In order to speed up Cassandra while handling spatial queries, we incorporate Stratio's Cassandra Lucene Index which holds spatial indexes. This gives same performance as using MongoDB and even better for some queries. MongoDB seems to handle aggregate query slower than Cassandra and PostgreSQL (e.g., region query involves two steps of aggregation).

Since we are using open source databases, the choice of which database to use depends merely on the needs and preferences, for instance MongoDB is well documented comparing to Cassandra. MongoDB uses XML language that is understood by internet, thus if one would like to work with different data traffic over internet MongoDB is a good choice. From developer perspective, it is easier to implement and integrate plugins to Cassandra than MongoDB. Cassandra seems to be updated every couple of weeks, this tick-tock releases are not immediately compatible with some plugin as it is the case in this paper, we have to use Cassandra 3.0.3 in order to be able to use Stratio's Cassandra Lucene Index 3.0.4, while at the moment, the current release is 3.4. One would choose to use PostgreSQL if relational database features is important to handle the data.

In terms of servers, if there is a constraint of number of servers, Cassandra is more preferable since it economically uses a less number of servers comparing to what MongoDB will require to provide same features.

APPENDIX

In tables 1, 2, 3, 4, 5, E.time is the average execution time of ten runs, Stdev is the standard deviation.

Table 1. Query processing time (in seconds) on 4 nodes installation (Dell powerEdge R320)

Query types	**Cassandra**		**MongoDB**		**PostgreSQL**	
	E. time	**Stdev**	**E. time**	**Stdev**	**E. time**	**Stdev**
Distance Q	0.036	0.077	0.024	0.011	0.79	0.0005
K-n Neighbors Q	0.029	0.005	0.024	0.0107	0.881	0.015
Range Q	0.008	0.008	0.021	1.83E-18	0.621	0.001
Region Q	0.045	0.011	1.562	0.030	1.221	0.001

Table 2. Query processing time (in seconds) on 3 nodes installation (Dell powerEdge R320)

Query types	**Cassandra**		**MongoDB**		**PostgreSQL**	
	E. time	**Stdev**	**E. time**	**Stdev**	**E. time**	**Stdev**
Distance Q	0.073	0.013	0.039	0.010	0.666	0.0005
K-n Neighbors Q	0.018	0.006	0.039	7.31E-18	0.886	0.015
Range Q	0.0130	0.024	0.04	0.011	0.666	0.001
Region Q	0.0515	0.021	1.593	0.030	1.222	0.001

Table 1. Query processing time (in seconds) on 2 nodes installation (Dell powerEdge R320).

Query types	Cassandra		MongoDB		PostgreSQL	
	E. time	Stdev	E. time	Stdev	E. time	Stdev
Distance Q	0.060	0.008	0.059	0.011	0.766	0.0008
K-n Neighbors Q	0.031	0.025	0.059	0.017	0.822	0.0007
Range Q	0.0147	0.002	0.045	7.31E-18	0.611	0.0006
Region Q	0.0518	0.024	1.633	0.030	1.225	0.001

Table 4. Query processing time (in seconds) on a single node installation (Dell powerEdge R320).

Query types	Cassandra		MongoDB		PostgreSQL	
	E. time	Stdev	E. time	Stdev	E. time	Stdev
Distance Q	0.017	0.076	0.001	2.2857E-19	0.789	0.0008
K-n Neighbors Q	0.012	0.007	0.001	2.29E-19	0.882	0.0007
Range Q	0.028	0.023	0.048	0.000422	0.621	0.0006
Region Q	0.054	0.019	2.526	0.066	1.225	0.0016

Table 5. Query processing time (in seconds) on a single node installation (Fujitsu RX600S5).

Query types	Cassandra		MongoDB		PostgreSQL	
	E. time	Stdev	E. time	Stdev	E. time	Stdev
Distance Q	1.121	0.001	1.579	0.298	2.243	0.181
K-n Neighbors Q	1.012	0.012	1.432	0.089	2.363	0.001
Range Q	1.432	0.001	1.654	0.068	2.154	0.001
Region Q	2.132	0.002	4.260	0.257	3.268	0.0009

ACKNOWLEDGEMENTS

This work is part of the research project "Scalable resource-efficient systems for big data analytics" funded by the Knowledge Foundation (grant: 20140032) in Sweden. We also thank HPI-FSOC, and Telenor Sverige.

REFERENCES

[1] S. Spaccapietra, C. Parent, M. L. Damiani, J. A. de Macedo, F. Porto, and C. Vangenot, “A conceptual view on trajectories,” Data Knowl. Eng., vol. 65, no. 1, pp. 126–146, 2008.

[2] Y. Zheng and X. Zhou, Computing with spatial trajectories. Springer Science & Business Media, 2011.

[3] N. Pelekis and Y. Theodoridis, Mobility data management and exploration. Springer, 2014.

[4] Jan, philip Matuschek, "Finding Points Within a Distance of a Latitude/Longitude Using Bounding Coordinates." [Online]. Available: http://janmatuschek.de/LatitudeLongitudeBoundingCoordinates#SQLQueries. [Accessed: 07-Mar-2016].

[5] R. Benetis, C. S. Jensen, G. Karčiauskas, and S. Šaltenis, "Nearest neighbor and reverse nearest neighbor queries for moving objects," in Database Engineering and Applications Symposium, 2002. Proceedings. International, 2002, pp. 44–53.

[6] E. Frentzos, K. Gratsias, N. Pelekis, and Y. Theodoridis, "Nearest neighbor search on moving object trajectories," in Advances in Spatial and Temporal Databases, Springer, 2005, pp. 328–345.

[7] M. Erwig, R. H. Gu, M. Schneider, M. Vazirgiannis, and others, "Spatio-temporal data types: An approach to modeling and querying moving objects in databases," GeoInformatica, vol. 3, no. 3, pp. 269–296, 1999.

[8] Y. Theodoridis, "Ten benchmark database queries for location-based services," Comput. J., vol. 46, no. 6, pp. 713–725, 2003.

[9] D. Pfoser, "Indexing the trajectories of moving objects," IEEE Data Eng Bull, vol. 25, no. 2, pp. 3–9, 2002.

[10] V. T. De Almeida, R. H. Güting, and T. Behr, "Querying moving objects in secondo," in null, 2006, p. 47.

[11] C. Düntgen, T. Behr, and R. H. Güting, "BerlinMOD: a benchmark for moving object databases," VLDB J., vol. 18, no. 6, pp. 1335–1368, 2009.

[12] L. I. Gómez, B. Kuijpers, and A. A. Vaisman, "Aggregation languages for moving object and places of interest," in Proceedings of the 2008 ACM symposium on Applied computing, 2008, pp. 857–862.

[13] Y.-J. Gao, C. Li, G.-C. Chen, L. Chen, X.-T. Jiang, and C. Chen, "Efficient k-nearest-neighbor search algorithms for historical moving object trajectories," J. Comput. Sci. Technol., vol. 22, no. 2, pp. 232–244, 2007.

[14] D. Pfoser, C. S. Jensen, Y. Theodoridis, and others, "Novel approaches to the indexing of moving object trajectories," in Proceedings of VLDB, 2000, pp. 395–406.

[15] K. Y. Besedin and P. S. Kostenetskiy, "Simulating of query processing on multiprocessor database systems with modern coprocessors," in Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on, 2014, pp. 1614–1616.

[16] R. Moussalli, I. Absalyamov, M. R. Vieira, W. Najjar, and V. J. Tsotras, "High performance FPGA and GPU complex pattern matching over spatio-temporal streams," GeoInformatica, vol. 19, no. 2, pp. 405–434, Aug. 2014.

[17] P. Huang and B. Yuan, "Mining Massive-Scale Spatiotemporal Trajectories in Parallel: A Survey," in Trends and Applications in Knowledge Discovery and Data Mining, Springer, 2015, pp. 41–52.

[18] R. Moussalli, M. Srivatsa, and S. Asaad, "Fast and Flexible Conversion of Geohash Codes to and from Latitude/Longitude Coordinates," in Field-Programmable Custom Computing Machines (FCCM), 2015 IEEE 23rd Annual International Symposium on, 2015, pp. 179–186.

[19] J. Han, E. Haihong, G. Le, and J. Du, "Survey on NoSQL database," in Pervasive computing and applications (ICPCA), 2011 6th international conference on, 2011, pp. 363–366.

[20] "What is Apache Cassandra?," Planet Cassandra, 18-Jun-2015. [Online]. Available: http://www.planetcassandra.org/what-is-apache-cassandra/. [Accessed: 23-Feb-2016].

[21] "CQL." [Online]. Available: http://docs.datastax.com/en//cassandra/2.0/cassandra/cql.html. [Accessed: 23-Feb-2016].

[22] tutorialspoint.com, "MongoDB Overview," www.tutorialspoint.com. [Online]. Available: http://www.tutorialspoint.com/mongodb/mongodb_overview.htm. [Accessed: 23-Feb-2016].

[23] "MongoDB for GIANT Ideas," MongoDB. [Online]. Available: https://www.mongodb.com/. [Accessed: 23-Feb-2016].

[24] J. Worsley and J. D. Drake, Practical PostgreSQL. O'Reilly Media, Inc., 2002.

[25] "Data distribution and replication." [Online]. Available: https://docs.datastax.com/en/cassandra/2.0/cassandra/architecture/architectureDataDistributeAbout_c.html. [Accessed: 25-Feb-2016].

[26] "Stratio/cassandra-lucene-index," GitHub. [Online]. Available: https://github.com/Stratio/cassandra-lucene-index. [Accessed: 23-Mar-2016].

[27] "Sharding Introduction — MongoDB Manual 3.2," https://github.com/mongodb/docs/blob/master/source/core/sharding-introduction.txt. [Online]. Available: https://docs.mongodb.org/manual/core/sharding-introduction/. [Accessed: 24-Feb-2016].

[28] "Distributed PostgreSQL, http://www.postgresql.org/docs/9.1/static/high-availability.html."

AUTHORS

Christine Niyizamwiyitira is currently a PhD student in Computer science at Blekinge Institute of Technology (BTH) in Sweden in Computer Science and Engineering Department. She completed her masters in 2010 in computer engineering from Korea university of Technology (KUT) in South Korea. She works at University of Rwanda as assistant lecturer. Her research interests includes Real time systems, cloud computing, high performance computing, Database performance, and Voice based application. Her current Research focuses on Scheduling of real time systems on Virtual Machines (uniprocessor & multiprocessor) and Big data processing.

Lars Lundberg is a professor in Computer Systems Engineering at the Department of Computer Science and Engineering at Blekinge Institute of Technology in Sweden. He has a M.Sc. in Computer Science from Linköping University (1986) and a Ph.D. in Computer Engineering from Lund University (1993). His research interests include parallel and cluster computing, real-time systems and software engineering. Professor Lundberg's current work focuses on performance and availability aspects.

INTENTIONAL BLANK

EXPLORING PEER-TO-PEER DATA MINING

Andrea Marcozzi - Gianluca Mazzini

Lepida SpA, Bologna, Italy
andrea.marcozzi@lepida.it - g.mazzini@ieee.org

ABSTRACT

The emerging widespread use of Peer-to-Peer computing is making the P2P Data Mining a natural choice when data sets are distributed over such kind of systems. The huge amount of data stored within the nodes of P2P networks and the bigger and bigger number of applications dealing with them as p2p file-sharing, p2p chatting, p2p electronic commerce etc.., is moving the spotlight on this challenging field. In this paper we give an overview of two different approaches for implementing primitives for P2P Data Mining, trying then to show differences and similarities. The first one is based on the definition of Local algorithms; the second one relies on the Newscast model of computation.

KEYWORDS

distributetd data mining; local algorithms; gossiping

1. INTRODUCTION

Open peer-to-peer networks have become very popular for several kinds of applications and they have emerged as an important paradigm for distributed computing, due to their potential for the involvement of millions of peers in the process of sharing and collaboration. One of the most interesting features of P2P networks is their ability for direct resource sharing among dynamic, decentralized client peers. Due to the improvement of the technologies relating networks connectivities, digital storages and devices, these kind of server-less networks storing a huge amount of varying data are growing very fast; mining such relevant amount of data can be of great importance for several purposes,[1] that's why investigating methods for P2P data mining has become a very interesting research field.

Traditional approaches see data mining systems download all the relevant data stored in a P2P network into a centralized location and then perform the classical data mining operation. In the scenario we have described above, this kind of solution results to be not always appropriate, hence Distributed Data Mining has been introduced.

---

[1]Commercial, scientific and medical purposes.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 165–174, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60613

Distributed Data Mining deals with data analysis in those environments in which data are distributed as for peer-to-peer networks and offers an alternative way to address this problem.

Researchers have developed several approaches for computing primitive operations (sum, average, max) on P2P networks. In this report we are going to introduce two different approaches: the first one is based on the concept of *local algorithms*[1], algorithms computing their results just with communications between immediate neighbors; the second one is based on the *Newscast model of computation* [4][3], a probabilistic epidemic protocol for information and membership dissemination.

In the next sections we will first give a brief overview on P2P data mining, its motivations and goals; then (section 3) we will introduce the concept of *local algorithms* [1] giving some examples; in section 4 we will introduce the *Newscast model* and give an idea of how it works along with some practical primitive implementations; finally in the last section we will draw some conclusions.

2. GOAL IN PEER-TO-PEER DATA MINING

One of the main goal of P2P data mining is to achieve as closer as possible the same results that can be obtained with a centralized approach without moving any data from the original location. That's why such algorithms must be highly scalable, tolerant to crashes and to peer “churn"[2] and mainly they must be able to calculate the results in-network instead of loading all the data in an unique system and then apply to itself the traditional data mining techniques.

As just said, there are several properties that are required by a peer-to-peer data mining algorithm:

scalability has been already mentioned and it is the foremost requirement; algorithms for peer-to-peer networks must be independent of the size of the network or at most they must be dependent of the log of the size;

anytimeness means that, since in some application the rate of the data change may be higher than the rate of computation, the algorithm must be able to provide a good and ad hoc solution at any time;

asynchronism is also crucial requirements for P2P algorithms: P2P networks are often huge, that means that any attempt to synchronize between the entire network is vane due to network latency or limited bandwidth;

decentralization means that the computations must be done in network, hence no centralized coordination must be used;

fault-tolerance is an other issue we have already mentioned; in a large P2P network can often happen that nodes crash or that they want to leave or join the network. That's why P2P algorithms

---

[2]By "Churn" we mean the situation in which some nodes leave the network and are suddenly replaced with brand new ones.

should be able to recover from these situations.

In the next sections we will present some primitive designed for Peer-to-Peer Data Mining purposes which show to comply with the above mentioned needs.

3. LOCAL ALGORITHMS

Approaches to P2P data mining have focused on developing data mining primitive operations over the network as well as more complicated data analysis algorithms.

Datta in [1] proposes some algorithms for calculating such primitives as sum and average, basing on the definition of *local algorithms*.

Given a constant k, for any network dimension, we can say that an algorithm is a local algorithm if there is a part of the input for which the algorithm terminates with communications expended per peer no greater than k and on the rest of the inputs, the communication expended per peer is of the order of the network size. Them can be divided into two categories: *exact local alg*orithm and *approximate local algorithms*: the former always terminate granting the same results that can be obtained with centralized methods; the latter instead, they can not give such level of accuracy.

Of course exact local algorithms can give better results, but it is not possible to develop them for every kind of situation.

In the next subsections will be given examples of both exact an approximate local algorithms.

3.1 Exact local algorithms (The Majority voting problem)

The Majority voting problem is a typical example of exact local algorithm which represents a situation in which each peer (P_i) of a P2P network has a number b_i which may be 0 or 1 and a threshold $1 > \tau > 0$ [3] (each node has the same τ). Peers want to collectively determine if the sum of all b_i is greater than $n\tau$, where n is the number of peers in the network.

Addressing this problem can serve as a primitive for several kind of data mining algorithms and can be used as a primitive for more complicated exact local algorithms.

P_i is a generic node in the network, N ei_i is one of its neighbors, C_i represents an estimate of the number of the nodes in the network and S_i is an estimate of the global sum. All the peers can only communicate with its neighbors and it is through this way that the just mentioned estimates are updated. We also define the threshold belief of P_i when such peer believes or not the majority threshold is met $(S_i - C_i\tau > 0)$.

Hence this threshold belief depends on the exchange of informations between neighbors. The crucial point of this approach concerns in deciding whether P_i needs to send a message to its neighbor P_j from which it has just received information on C and S. P_i will not send such a message if and only if it can be certain that such information cannot modify the threshold belief

---

[3] Actually in the original paper [1] was just indicated $\tau > 0$

of P_j . On the other end, if it cannot be certain of this, a message must be sent. This decision is taken on the basis of the estimate P_i makes on the values of C_j and S_j together with its values (C_i and S_i). When a node P_i decides to send a message, then it sends all of its informations about S and C, excluding those sent from P_j .

This approach is considered to be robust to data and network change: when a peer P_i changes its data, then P_i recomputes C_i and S_i and applies those conditions to all of its neighbors; if a peer P_j leaves the network, then P_i recomputes S_i and C_i without taking into account the informations from P_j .

Discussion

Exact local algorithms can be very useful in solving data mining problems in P2P networks but unfortunately they are very limited. The scope of such algorithms is restricted to functions that can have a local representation in the given network and they are limited to those problems which can be reduced to threshold predicates (as for the majority voting problem). An example of application is given in [1], where an exact local algorithm (based on the majority voting problem) is used for monitoring a K-means clustering of data distributed over a peer-to-peer network. In this application K-means clustering is performed in the traditional way (on a centralized system). After this, results are sent to the peers of the network and the local algorithm for monitoring the K-means clustering is executed: this algorithm just raises an alert if the centroids needs to be updated.

It is quite impossible to develop an exact local algorithm to compute the average of a set of data distribute over the network, that is why it is impossible to solve some data mining problems with exact local algorithms (as for example P2P K-means clustering). For addressing this, two different mechanism are proposed: the first one (section 3.2) describes an approximate local algorithm to perform the K-means clustering over a P2P network [2]; the second one (section 4) describes the Newscast model of computation for calculating means over data distributed on P2P overlay networks.

3.2 Approximate local algorithms (P2P K-means clustering)

The P2P K-means clustering algorithm [2] is an iterative algorithm requiring only local communications and synchronization at each iteration: nodes exchange messages and synchronize only with their neighbors. The goal is for each node to converge on a set of centroids that are as close as possible to the centroids that would have been produced if the data from all nodes was first centralized, then K-means was run.

The algorithm is initiated with a set of starting centroid selected at random. P_1, P_2,...., P_n denote the nodes in the network and X^i denotes the data set held by each node; the global data set is denoted as a X which equals to $\bigcup_{i=1}^{n} X^i$; the list of neighbors of a generic node i is denoted by $Nei^{(i)}$. Each node stores: a set $\{w_{j,k}^{(i)} : 1 \leq j \leq K\}$ indicating the centroids (*local centroids*) held by the node *i* at the beginning of cycle *l*; a termination threshold $\gamma > 0$; and a cluster count $|(w_{j,l}^{(i)})|$ which is the number of tuple in X^i for which $w_{j,l}^{(i)}$ is closer than any other $w_{h,l}^{(i)}, h \neq j$.

Each iteration of the algorithm is divided in two steps: the first one is similar to the centralized K-means in which peer P_i assigns each of its points to the nearest centroid; in the second one peer P_i sends a poll message to its immediate neighbors and waits for a respond. This request consists of a pair $\langle k, l \rangle$ (id, current iteration number) which is done in order to make the neighbors to respond with their local centroids and cluster count for iteration *l*. Once they have all responded, P_i updates its j^{th} centroid at the beginning of iteration $l + 1$. The update is a *weighted average*[4] of the local centroids and counts received from all immediate neighbors (for their iteration *l*). Then it moves to the next iteration of the K-means algorithm and repeats the whole process. If the maximum change in position of the new centroid after an iteration remains above the defined threshold γ, then P_i goes on iteration $l + 1$.

The key point is how do the peers respond to those requests. Suppose peer P_i receives a poll message $\langle k, \hat{l} \rangle$ from node k at its iteration $\hat{l}$: if $\hat{l} < l$, P_i sends its local centroid and cluster count to peer P_k; if $\hat{l} > l$, that means P_i does not have local centroids for iteration $\hat{l}$ and in such case, the poll message is put in the poll table of P_i; if $\hat{l} = l$, P_i checks if it contains local centroids and cluster counts for P_k, if so, they are sent to P_k, else the poll message is put in the poll table. Finally P_i will check its poll table at each iteration and will respond to any message it can.

At the end of each iteration *l*, if no important changes are detected on the cluster centroids (the maximum change in position is below the user defined threshold γ). each node could enter a termination state. In that case, such peer no longer updates its centroids and sends poll messages. However, it does responds to polling messages from its neighbors. Once all peers enter into the terminated state, the algorithm is terminated.

Experiments results and discussion

The P2P K-means clustering algorithm [1] presented in the previous section is a very important example which gives the idea of how is important to implement good primitives for data mining in distributed environments. The algorithm is in fact based on a primitive derived from the majority voting problem which is a primitive developed for peer-to-peer systems.

Datta in its works [1][2] performed several experiments with its P2P K-means clustering algorithm which seems to achieve good results. Experiments were conducted in both static and dynamic environments with a network of 1000 nodes: in both cases was calculated the accuracy with respect to the classic centralized K-means and the communication cost. In the static environment high accuracy is found (less than 3% of points per node misclassified on average) while no significant impact on this has had the method of assigning data points to node (uniformly or non-uniformly). However this has had a significant impact on communication costs: the number of bytes received per node increases slowly with network size for uniform assignment; the cost increases more sharply for non-uniform assignment.

Experiments were also conducted on a dynamic environment (with nodes leaving and joining the network) and even here good accuracy was found (less than 3:5% misclassified on average) which remains stable when the network evolves. Even increasing the network size did not seems to change the accuracy significantly: this proves that the algorithm is highly scalable.

[4]Such weighted average is calculated by primitive implemented with local algorithms.

4. DATA MINING THROUGH THE NEWCAST MODEL OF COMPUTATION

As already said, researchers working on distributed data mining have been focusing on investigating techniques for calculating primitives as *average*, *maximum* etc.. in distributed environment. Even Kowalczyk and Jelasity in [4] focused on this, but their approach is slightly different from the Datta's one [2].

First of all they adopted two important constraints: the first is that all the nodes (peers) store as few as one single data instances (in [1] each peer held several data instances); the second is that there is practically no limit on the number of nodes (even in [1] there was no potential limit on this, but experiments were always conducted on small networks of the order of thousand nodes). Furthermore, as in [1] nodes can leave and join the network as in a dynamic environment. For this, in this approach are needed resources that scale directly with the size of the network, which is a feature distinguishing it from local algorithms.

In the next subsections we are going to first introduce the Newscast model of computation, then we will propose some primitive for distributed Data Mining within this model and finally we will draw some conclusions.

4.1 The Newscast model (an overview)

Newscast [3] is a *gossip-based* topology manager protocol. Its aim is to continoulsy rewire the (logical) connections between hosts. The rewiring process is designed in such a way that the resulting overlay is very close to a random graph. The generated topology is thus very stable and provides robust connectivity.

As in any large P2P system, a node only knows about a small fixed set of other nodes (due to scalability issues), called *neighbors*. In Newscast, the neighborhood is represented by a partial, fixed c size view of node *descriptors* composed by a node address and a logical *time-stamp* (e.g., the descriptor creation time).

The protocol behavior performs the following actions: selects first a neighbour from the local view, exchanges the view with the neighbor, then both participants update their actual view according to the received view. The data actually sent over the network by any Newscast node is represented by the node's own descriptor plus its local view.

In Newscast, the neighbor selection process is performed in a random fashion by the SELECTPEER() method. The UPDATE() method is the Newscast core behavior. It merges ($\bigcup$ operation) a received view (sent by a node using SENDSTATE()) with the current peer view in a temporary view list. Finally, Newscast trims this list to obtain the new c size view. The node descriptors discarded are chosen from the most \old" ones, according to the descriptor time-stamp. This approach changes continuously the node descriptors hold in each node view; this implies a continuous rewiring of the graph defined by the set of all node views.

The protocol always tends to inject new informations in the system and allows an automatic elimination of old node descriptors using the aging approach. This feature is particularly desirable to remove crashed node descriptors and thus to repair the overlay with minor efforts.

Newscast is also cheap in terms of network communication. The traffic generated by the protocol involves the exchange of a few hundred bytes per cycle for each peer.

Figure 1: Conceptual model of the collective of agents and the news agency (from [4]).

Trying to talk about this model on an higher level, we can say that it is based on two main concepts: the collective of agents and the news agency (see Figure 1). The agents communicate through the news agency. Although the news agency plays the role of a server orchestrating the communication schedule, it is a virtual entity implemented in a fully distributed P2P solution. The communication schedule is organized into cycles and in each of them the news agency collect exactly one news item from all the agents. At the same time it prepares for each agent a randomly chosen set of news item from the previous cycle and delivers these set to the agents. In the next subsection we present an example of averaging primitives implemented within the model.

4.2 Basic Statistics

The ability of calculating the mean is central for implementing some basic data mining algorithms in Newscast. As we said the Newscast communication schedule is organized into cycles so what we want is an algorithm able to calculate in few cycles the average of the values held by the nodes of the Newscast Network. In this section we present three averaging algorithms developed on Newscast.

4.2.1 Basic Averaging (BA)

The *Basic Average algorithm* [4] is the simplest way to achieve this. During the first cycle each agent publishes its value so that the news agency get a copy of all the values that must be averaged. Next all agents whenever they receive news, they make the average of these values and then publish it. An important observation must be done: in every cycle the news agency receives a set of values that on average has the same mean of the original set, but the variance will be getting smaller and smaller with the number of cycles. This is the most important result of this algorithm. From the experiments performed, it can be shown that after k iterations of the "averaging operations", the variance drops to $(1 - 1/n)^k$ of its initial value.

As already said this is probably the simplest averaging algorithm that we can think on the Newscast model, but its simplicity pays the lack of adaptation. In fact if we think to a network where nodes leave and join, change their values and where nodes can temporary or permanently crash, the BA is not able to fit with these dynamical needs. To address this, the *Systematic Averaging algorithm* [4] is proposed (next section).

4.2.2 Systematic Averaging (SA)

The SA algorithm [4] achieves adaptation by constantly propagating agents current values and temporal averages through the news agency. Therefore, any change in the incoming data will quickly affect the final result.

A small positive integer d is fixed and it is used to control the depth of the propagation process. The news items are vectors of $d + 1$ elements: the first element of a news item X is x_0 and contains the agent value (called 0-order estimate of the mean); x_1 is the means of two 0-order estimates and it is called 1-order estimation mean; x_d, which is the last element, is the average of two estimates of order d -1 and is called an estimate of order d. In this way consecutive elements of X will be *balanced*. The result of this propagation is represented by x_d.

Even the SA algorithm has the ability to drop the variance: it decreases in an exponential way. Moreover the system can react to changes in the input data within d iterations.

4.2.3 Cumulative Averaging (CA)

The two algorithm we have just seen have the ability of reducing the variance very fast, but, due to the randomness characterizing the Newscast engine, the output value could be different from the true mean.

This problem is solved by the CA algorithm [4]. It runs two processes in parallel: in the first one agents updates their estimate of the mean of the incoming data, while in the second one the mean of these estimates is collectively calculated by a BA procedure.

4.3 Experiments configurations and results

The experiments we are going to describe [4] relates to the three averaging algorithms we have just mentioned. These are based on tests with different network sizes (from 10000 to 50000) and different data set. For each configuration were executed 100 independent runs. Were also used three different kind of data sets: Gaussian, where the value of each agent was taken independently from a Gaussian distribution; half-half, where half of the agents had value 0 and the other half 1; and peak where all but one agents hold the value 0.

With respect to the convergence rate the BA algorithm was fastest (20-30 iteration), the SA was slower (50 iterations) and the CA was the slowest (100 iterations). With respect to accuracy, the situation was inverted: BA was worst, SA better and CA best. The deviation from the true mean depends on the used distribution. Peak distribution has the intent of showing the "true power of the averaging algorithm", in fact we can note from the results with such distribution that the mean and variance with BA are respectively 0.935 and 0.656, while with SA they are 0.98 and 0.265.

4.4 Applications

As already seen with the primitives calculated with local algorithms, even here the averaging primitives implemented through the Newscast model can be used in several data mining tasks as for example for classification techniques. In [4] is reported an example in which the above mentioned averaging algorithms, with a little modification, are used for finding the Naive Bayes classifier for data that are arbitrarily distributed among the nodes in a P2P Network.

Kowalczyk *et al*. have implemented the Naive Bayes algorithm using BA as the base averaging method. As they had to maintain estimates of several means, they had to represent news items by vectors of the same length as the number of the means and to run BA on all coordinates at the same time. They have then tested the performances of that algorithm performing several experiments and then comparing the results with a classical Naive Bayes centralized algorithm. In most cases, although the model parameters were slightly different, no difference in the classification rate were found.

Most statistics that are used by other classification algorithms are defined in terms of ratios (or probabilities) that have the same form as described above. Consequently they can be implemented within the newscast framework.

5. CONCLUDING DISCUSSION

In this paper we have described in brief two interesting approaches to Data Mining in Peer-to-Peer Systems. The first one from Datta et al. [1] is based on the concept of *local algorithms* and the second one from Kowalczyk *et al*. [4] uses the *Newscast model of computation*.

Both the authors are intent to supply techniques for calculating primitives for P2P networks, primitives that form the basis for more complicated Data Mining algorithms. Both the approaches result very interesting even though they differs in some aspects.

Calculating primitives through the Newscast model of computation, resulted a winning strategy. The main task the authors wanted to deal with, was seeking a model for data spread over a number of agents; this was addressed through Newscast which is based on an epidemic protocol for disseminating information and group membership. The fact that the model lies on such robust and highly scalable protocol, states the goodness of the model itself, which inherits all these good features.

One important thing the two approaches have in common is that in both cases peers communicates only with their immediate neighbors but the second one (Newscast) does this in an epidemic-style manner so that results can be spread very quickly and all the agents can hear about the final solution in a short amount of time.

An important difference between the Newscast model and local algorithms which derives from what we have just said, is that in the Newscast model the terminations is reached once all agents have heard about the final results: although there is no signal that informs the agents that the result is found, using the theory of epidemic algorithms (which works as a broadcasting mechanism), all agents will hear about the final solution very quickly. Local algorithms instead, they terminates once a certain threshold is met: when an agent has reached an user defined

threshold (in the P2P K-means clustering it related the change in position of the centroids), it enters the terminated state; once all agents have reached such state, the whole process stops.

An other main dfference between the two approaches is that the News cast model requires resources that scale directly with the size of the network. So the resources required by the algorithm are dependent from the size of the system. In spite of this, the model results very scalable and robust. Local algorithms instead, computes their results using informations from a handful of nearby neighbors which leads to a good level of scalability too. It has also been proved that they are very good at adjusting to failure and changes in the input locally (see section 3.1). Even with the Newscast model it is possible to achieve these properties: we have seen the *Systematic Average algorithm* which is able to adjust to changes in the value of the agents on-the-fly and we have also seen that the tendency of the protocol to insert new informations in the system, allowing an automatic elimination of old node descriptors, is particularly desirable to remove crashed node descriptors and thus to repair the overlay with minor efforts.

In light of this, we can't say if one of the two approaches is better than the other one. We can certainly state (basing on the provided results) that both are able to fit with the peer-to-peer networks requirements: they are highly scalable, robust to node crashes and data set changes, decentralized and asynchronous. Once applied to real P2P Data Mining algorithms as K-means clustering and Naive Bayes classification, they have also shown a good level of accuracy and convergence to the results obtained with traditional centralized techniques.

In spite of this, data analysis in P2P systems still offers lot of challenges for the researchers. The experiments on the primitive and the distributed data mining algorithms we have described in this report, have shown good results but they come from lots of simulations done on P2P networks testbeds, hence we have “no mathematical proofs" of their absolute validity. As future work, it would be interesting and challenging at the same time to test this approaches on a platform like PlanetLab (a reliable testbed for overlay networks)[5], or even better on a real Peer-to-Peer Overlay Network.

REFERENCES

[1] Datta, S., Bhaduri, K., Giannella, C., Wol, R. (2005) Distributed Data Mining in Peer-to-Peer Networks. Invited submission to the IEEE Internet Computing special issue on Distributed Data Mining.

[2] Datta, S., Giannella, C., Kargupta, H. (2006) K-Means Clustering Over a Large, Dynamic Network. Accepted paper in SIAM2006 Data Mining Conference.

[3] Jelasity, M., van Steen, M. (2002) Large-Scale Newscast Computing on the Internet. Internal Report IR-503, Vrije Universiteit Amsterdam, Department of Computer Science, Amsterdam, The Netherlands.

[4] Kowalczyk, W., Jelasity, M., Eiben, A. (2003) Towards Data Mining in Large and Fully Distributed Peer-to-Peer Overlay Networks. Technical Report IR-AI-003, Vrije Univeriteit Amsterdam, Department of Computer Science, Amsterdam, The Netherland.

---

[5]http://www.planet-lab.org, Last visited March 2016

ENHANCED PROTOCOL FOR WIRELESS CONTENT-CENTRIC NETWORK

Chan-Min Park[1], Rana Asif Rehman[2], Tran Dinh Hieu[2], Byung-Seo Kim[3]

[1]Graduate School of Smart City Science Management,
Hongik University, Sejong, Republic of Korea
walkinpcm@gmail.com
[2]Dept. of Electronics and Computer Engineering,
Hongik University, Sejong, Republic of Korea
asifrehman7@gmail.com, trandinhhieu1989@gmail.com
[3]Dept. of Computer and Information Communication Engineering,
Hongik University, Sejong, Republic of Korea
jsnbs@hongik.ac.kr

ABSTRACT

Recently, Content-Centric Networking (CCN) was introduced and is expected as a new concept of future internet architecture. Even though CCN is initially studied for wired networks, recently, it is also studied for wireless environment. In this paper, we discuss improvement method for efficient content flooding over wireless CCNs. The proposed scheme of this paper use MAC Address of nodes when Interest and Data Packet are forwarded in order to limit the area of flooding of packets. The proposed protocol not only reduces the spread of Data packets, but also offers priority of forwarding to nodes of shortest path. As a consequence, it reduce content download time which is proved by extensive simulations.

KEYWORDS

Network Protocols, CCN, Future Internet, Protocol, Wireless.

1. INTRODUCTION

As mobile communication devices such as smartphones are widely used and rapidly spread out, so that peoples are easily connected through Internet, which is so-called Internet-of-People (IoP). Furthermore, such IoP not only increases tremendous of individuals data traffics, but also requests lots of huge contents such as video and music from content servers. Even though current Internet architecture provides efficiently end-to-end or host-to-host communications, it has been troubled to deal with user's mobility environment and services of aforementioned massive contents provided to unspecified number of users. Particularly, due to popularity of video streaming services such as TED and YouTube, more video-like content traffics are flooded over the networks. In such video streaming services, many users request even same content in a different time to same content server which might be located far from their locations. For example, let's assume that there is content server in U.S. and there are 10 people in Korea to

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 175–183, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60614

request the content in a different time and different area. In this case, the content server will send same content 10 times from U.S. to Korea. This is very inefficient way to provide content. In order efficiently to request and provide the content itself, Content Centric Network (CCN) concept has been proposed in year 2009 by Van Jacobson [1]. In CCN, when the first content is delivered, we allow some server or router in Korea to store the content. Therefore, the second user's request on the same content can be provided from the nearby server or router, not far from server in U.S. Furthermore, CCN proposes to use a content name to request/distribute content instead of using IP address. That is, while current Internet focuses on "where to get" using IP address, CCN focuses on "what to get" using content name itself.

Even though initial studies for CCN have been conducted on the wired networks, recent studies moves to CCN-based wireless networks and is actively being conducted [2]-[10] because as mentioned earlier, people enjoys contents through smart handheld devices while they are moving. However, the research on wireless CCN is still early-state and there are many issues to be resolved. One of issue is long delay to download contents because of broadcasting and flooding-based transmissions adopted by wireless CCN. In this paper, we focus on how to reduce flooded traffics in wireless CCN in order to reduce download time of contents. To resolve the issue, we propose a novel protocol to reduce the number of content request messages, named Interest packet, flooded over the networks. As a result, the proposed protocol reduces collisions and wasted time in wireless CCN.

In Section 2, the fundamentals of CCN and some prior arts related with wireless CCNs are introduced. In Section 3, after motivation of this research is introduced, the proposed protocol is described. In Section 4, the proposed protocol is evaluated in terms of the content download time and conclusions are made in Section 5.

2. PRIOR ART

CCN is composed of three devices and three packets [1]. Three devices are consumer, provider, and intermediate nodes, and three packets are Interest, Data, and Announcement packets. Each device consists of three components: Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB). A consumer requests a content by sending an Interest packet and the Interest packet is flooded over a network by all intermediate nodes. Once receiving the Interest packet, a node to check its CS if it has the content. If it has, it sends Data packets in the manner of broadcasting and it becomes a provider. If it doesn't, it records the information of Interest packet in its PIT and broadcasts the Interest packet to Networks. Therefore, any node that has request content can be a provider which is a content server in terms of conventional IP-based networks. When receiving a Data packet, then it checks PIT to ensure any node requested the content before. If yes, it broadcasts the Data packet. Otherwise, it discards the packet. In CCN, a provider can broadcast announcement packet to let nodes in network know specific content that the node has in CS.

After the concept of CCN was proposed, studies on wireless CCN have been actively performed [2]-[10]. Enhanced-Content-centric multiHop wireless NETwork (E-CHANET) is designed for wireless networks adopting CCN concept [2]. In E-CHANET, Announcement packet and FIB component are removed from the system. Instead, to cope with erroneous wireless channel, it uses Interest-Data-based two-way handshake performed between consumer and provider. Therefore, every Data packet transmission is requested by one Interest packet. If there is a lost

Data packet, same Interest packet is retransmitted. Calculating wait-time for receiving the requested Data packet and handoff method between different providers are also designed. The CCN communication method is also applied to vehicular networks [4]. CarSpeak enables a car to query and access sensory information captured by other cars in a manner similar to a way to request a content in content-centric approach. Authors in [4] try to solve packet loss problem caused by node mobility in CCN-based MANETs. In [4], both of broadcast and unicast are used for the transmissions of Interest packets, and additional function is added which is to validate if the next hop node is available when a node receives Interest or Data packet. Flooding issue in CCN has been studied in [5][6]. In [5], the shortest hop count is found by flooding Interest packets and then Data packet is flooded back to consumer within the hop count. In [6], the authors proposed a new scheme called Neighborhood-Aware Interest Forwarding (NAIF) for NDN-based mobile ad hoc networks. This scheme basically reduces the flooding overhead in the network. In which, the intermediate node locally decide to propagate or drop the received Interest packet based on the forwarding rate. Authors in [7] propose a scheme for multihop based wireless CCN. Two additional packets, called EFS-ACK and EFS are also utilized in addition to the Interest and Data packets. In [8], the authors present an energy aware forwarding scheme for multihop wireless ad hoc networks. In which, the packets are forwarded based on node's residual energy. Authors in [9] propose forwarding strategies (i.e. BF, PAF) for wireless CCN ad hoc networks. These schemes are deeply analyze and well evaluated. Furthermore, the authors also highlight the advantages and shortcoming of both approaches. Kim et al., in [10], present a novel scheme, named AIRDrop in which communication is based on unicast manner. Proposed AIRDrop scheme also takes into account the extra tables and buffers during its communication operations.

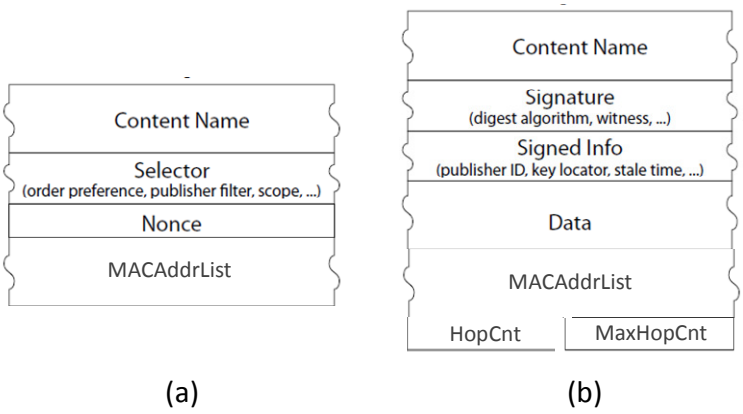


Figure. 1. Revised packet formats for the proposed protocol: (a) Interest packet and (b) Data packet

3. PROPOSED PROTOCOL

3.1. Motivation

Although previously proposed protocols for CCN-based wireless networks provide advantages for delivering contents, their performance is still struggling in terms of content download time. One of the reasons is inefficient Data packet forwarding mechanism. In wireless environments, using broadcasting manner, the direction of packet flows cannot be controlled, so that too much flooding causes delays to download contents. Since many wireless routing protocols uses unicasting and IP addresses, so that packet's flow can be controlled using IP addresses. However, because wireless CCNs do not use IP address and use broadcasting address for MAC protocol, it

is hard to restrict flooding area and control the content flows. On the other hand, totally removing flooding in wireless CCNs also removes advantage of CCNs' nature which is to spread contents over the networks for possible upcoming content request. Therefore, the objective of this paper is to propose a protocol for wireless CCNs which not only limits flooding of Data packets in a certain range over the networks, but also speeds up content download time using a shortest path from a provider to a consumer.

In the proposed protocol, the way to forward Interest packet is same as that of E-CHANET. However, whenever nodes forwards the packet, they add their MAC addresses in the packet. When the list of MAC addresses of the Interest packet arriving firstly at a provider, it will be used as the shortest path for Data packet's forwarding. Furthermore, since nodes in the list are allowed to have priorities to access the channel, Data packets can be arrived at a consumer faster comparing to the conventional wireless CCNs. Unlike common unicasting routing protocols used in wireless ad-hoc networks, the Data packets is not forwarded only by the nodes in the list. The packet is also forwarded in some area of the nodes in the list. That is, the Data packets are flooded in a restricted area around the shortest path. Therefore, it achieves not only a better content download time, but also flooding of Data packets.

3.2. New Frames

In the proposed protocol, Interest and Data packets' formats of E-CHANET are modified as shown in Fig.1. As shown in Fig. 1(a), for the Interest packet of the proposed protocol, one field, called *MACAddrList*, is added at the end of the packet. This field includes MAC addresses of intermediate nodes between a consumer and a provider. When a node receives an Interest packet and decides to forward it, it adds its MAC address in the field. Therefore, MAC addresses are accumulated in the field as the Interest packet is forwarded to a Provider, so that the size of the field is varied according to the number of nodes forwarding the Interest packet before it arrives at a provider. This filed is also used for Data packet as shown in Fig. 1(b). The list of addresses in the field of Interest packet is copied to the field of Data packet by a provider before the Data packet is transmitted. While a node adds its MAC address into *MACAddrList* field when receiving an Interest packet, it removes its MAC address from the field of Data packet when it receives the data packet and its address is in the field. *MaxHopCnt* is the number of MAC addresses indicating how many nodes the Interest packet have come through from a consumer to the provider. *HopCnt* field in Data packet indicates how many hops the Data packet can be forwarded. That is, this limits forwarding Data packet unnecessary further. When a node receives a Data packet, it recalculates *HopCnt* which is explained in detail in Section 3.3. *HopCnt* is differently calculated depend on whether or not the node itself is in *MACAddrList*.

3.3. Operation of the proposed protocol

In this section, operations of the protocol are explained in two parts: cases when a node receives Interest packet and Data packet.

Process when a node receives an Interest packet.

The procedure of the proposed protocol when a node receiving an Interest packet is shown in Fig. 2.

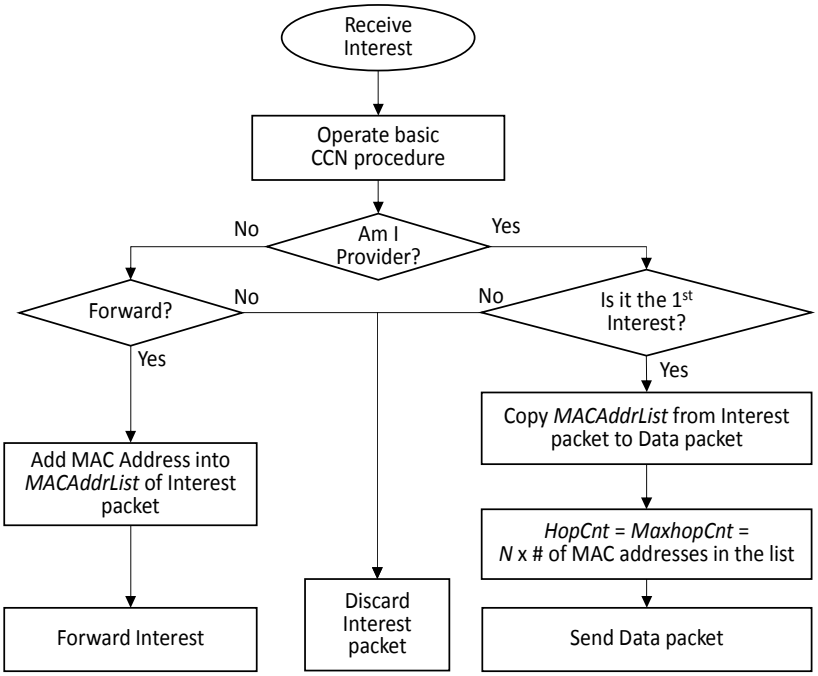


Figure. 2. Procedure of the proposed protocol when a node receives Interest packet

When a node receives an Interest packet, it performs the process defined from conventional wireless CCN such as E-CHANET. That is, it checks CS and PIT and decides whether or not to forward the packet. If the node acts as a relay node and the received Interest packet needs to be forwarded, it add the last 24bits of its MAC address to *MACAddrList* field in the received Interest packet and forwards the packet to the neighbours. If the packet is already received before, the node discards the packet.

If the node becomes a provider (that means it has the requested contents in its CS), it checks if it already received an Interest packet requesting same content and nonce number. If it did, it discards the packet. Otherwise, it records the first packet requesting the content with the nonce is received, and then at first, it copies the list in *MACAddrList* field in the Interest packet to *MACAddrList* field in Data packet to be sent. Secondly, it set *HopCnt* field to N times the number of MAC addresses in the field. Finally, it sends the Data packet to Consumer.

In the process, when a provider receives an Interest packet, it sets *MaxHopCnt* to N x the number of MAC addresses in *MACAddrList* of the Interest packet.

Process when a node receives a Data packet.

When a node receives Data packet, it processes conventional CCN process. Then it checks if it need to forward the received Data packet. If it does not need to forward the packet, it discards the packet. If it need to forward it, but *HopCnt* is 0, it discards the packet. Otherwise, the node checks if there is its own MAC address in the *MACAddrList* of the Data packet. If there is, *HopCnt* of the Data packet is set to *MaxHopCnt* and the node's own MAC address is removed from *MACAddrList* of the packet. In addition, after choosing random number between 0 and W_s,

it sets its deferring time, *DeferT*, to the random number x slot time, *SlotT*. After waiting *DeferT*, it forwards the Data packet to neighbours. If there isn't its own MAC address in the *MACAddrList* of the Data packet, the node checks if *HopCnt* of the packet equals to 1. If it is, *HopCnt* is set to 0. Otherwise, *HopCnt* is set to the integer that is not higher than root of *HopCnt* of the packet. After choosing random number between 0 and W_n, it sets its *DeferT*, to the random number x *SlotT*. After waiting *DeferT*, the packet is forwarded to the neighbours.

The proposed protocol provides two unique characteristics to efficiently forward Data packets. The first characteristic is to limit the Data packet's flooding by exponentially reducing *HopCnt* as explained in Step 4-2a. If a node receiving the Data packet coming from the non-shortest path, the Data packet is forwarded within number of hops. If it is coming from the shortest path, its Data packet is flooded as many hops as *MaxHopCnt* as explained in Step 4-1a. Therefore, it maintains an advantage of conventional CCN's flooding as well as prevents from unnecessary packet flooding. The second characteristic is to give transmission priority to nodes on the shortest path as explained in Step 4-1c and 4-2c. In conventional wireless CCNs, since all data packets are broadcasted in MAC payer, there is no retransmissions and backoff time increase. Because of this, it causes lots of collisions due to concurrent transmissions. Therefore, to resolve concurrent transmissions, CCN layer provides Deferred Time which a node waits random time before transmission. In this case, to give better transmission opportunity to the nodes along with the shortest path, the proposed protocol allows them to choose random number from the smaller range of numbers (0~Ws) than other nodes. The other nodes choose the random number from the wider range of numbers (0~Wn). As a result, nodes on the shortest path have the higher transmission opportunities than others.

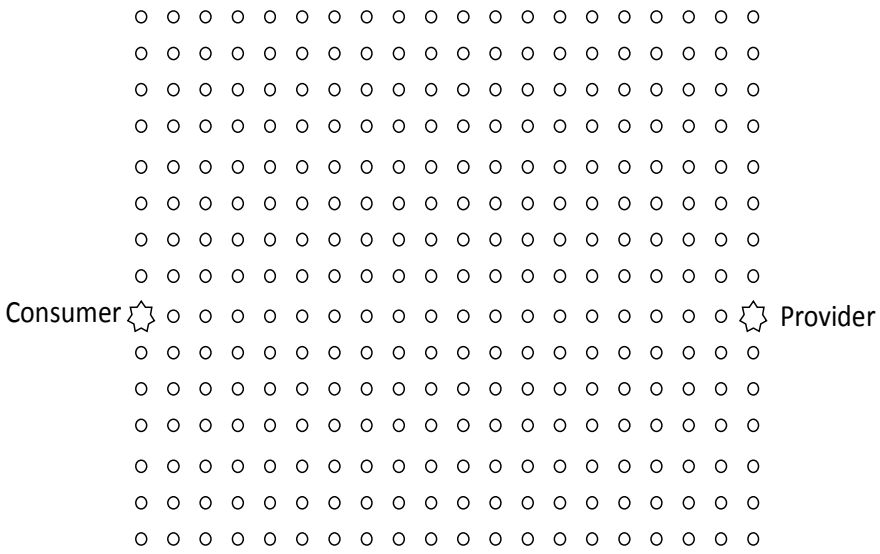


Figure. 3. Network Topology for the simulations

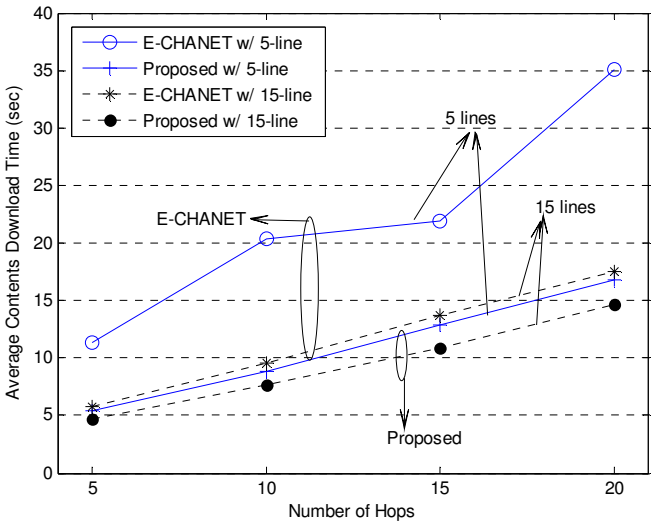


Figure. 4. Average content download time as a function of the distance between a consumer-provider pair

4. PERFORMANCE EVALUATIONS

In this section, the proposed protocol is comparatively evaluated with E-CHANET-based protocol. For evaluation studies, we have used ndnSIM [10] software module that is based on Network Simulator-3 (NS-3) version 3.16. In the simulations, one content consists of 100 packets and payload length of a data packet is set to 1200 bytes. Physical layer transmission rate is set to 6Mbps and 5GHz-carrier frequency-based IEEE802.11a standard is adopted. Medium access control protocol is based on IEEE802.11e standard. For the channel model, log normal path loss model with path loss exponent 3 is used. The simulations results are collected from 100 times simulations and one simulation completed when a consumer receives all 100 chunks packets from a provider.

The performances are evaluated as functions of network sizes by varying distances of a consumer-provider pair and the number of lines of Y-axis in grid topology. The topology for the simulations are shown in Fig. 3. Fixing a consumer's location, performances are evaluated varying distances to a provider. The distances are 5, 10, 15, 20 hops. In addition to varying distances between the consumer and the provider, the grid sizes are varied as like 5 and 15 lines. 1 line means there is only one path between the consumer and the provider. That is, the number of hop indicates the number of relay nodes in x-axis over grid topology while the number of lines indicates the number of relay nodes in y-axis in the topology. Therefore, we evaluates the performances over 8 sizes of networks (4 types of hops and 2 types of lines).

Fig. 4 shows the average content download time as varying the distance between a consumer and provider when the numbers of lines are 5 and 15. As mentioned before, the distance is represented using the number of hops between the consumer-provider pair and the distance is varied from 5 to 20 hops. As shown in the figure, the proposed protocol reduces the download time from 16% to 52% comparing to E-CHANET protocol. From the figure, as the distance increases, the reduction of download time is larger. The reason is because E-CHANET protocol makes more packets flooded over the networks as the distance increase. Since higher traffics over

the networks causes higher download time because of packet losses due to collisions and longer backoff time in MAC layer. It is noticed that there is exceptionally long download time of E-CHANET protocol in 5 lines in Fig. 4. This can be explained from Fig. 5 showing the average number of retries. As Fig. 5 shows, E-CHANET protocol has the highest number of retries. Since the forwarding path becomes narrower, the advantage of E-CHANET protocol's flooding-based Data packet forwarding decreases. Even though flooding-based packet forwarding method cause lots of collisions because too many nodes are participating in transmissions, the packet can eventually delivered to the consumer through many alternative paths because it is flooded. However, in the narrower network like 5 lines, the number of alternative paths are limited, so that the advantage of flooding decreases. Therefore, the number of retransmissions increases which leads the download time's increase.

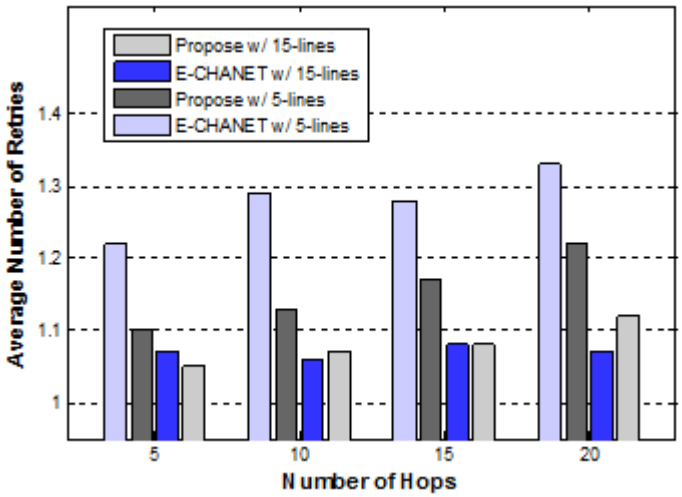


Figure. 5. Average number of retries as a function of the distance between a consumer-provider pair

5. CONCLUSIONS

In this paper, a novel protocol for wireless CCNs is proposed. Even though the protocol uses broadcasting and flooding-based Data packet forwarding, by restricting flooding area and giving transmission priority to a nodes on the shortest path, the protocol resolves the long content download time issue. The proposed protocol is evaluated throughout extensive simulations and it is proved that the objective of the proposed protocol successfully achieved by reducing a content download time up to 50% comparing to the conventional protocol.

ACKNOWLEDGEMENTS

This research was supported in parts by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2013R1A1A2005692) and in parts by the International Science and Business Belt Program through the Ministry of Science, ICT and Future Planning (2015K000270).

REFERENCES

[1] V. Jacobson, D.K. Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, and R.L. Braynard, “Networking named content,” Proc. Int. Conf. on Emerging Networking Experiments and Technologies, (CoNEXT’09), Rome, Italy, Dec. 2009. pp. 1-12.

[2] M. Amadeo, A. Molinaro, and G. Ruggeri, "E-CHANET: routing, forwarding and transport in information-centric multihop wireless networks," Computer Communications, vol. 36, no. 7, pp. 792-803, April 2013.

[3] M Amadeo, C Campolo, and A Molinaro, "Enhancing content-centric networking for vehicular environments," Computer Networks, vol. 57, no. 16, pp. 3222-3234, November. 2013.

[4] O. Adem, S. Kang, Y.B. Ko, "Packet Loss Avoidance in Content Centric Mobile Adhoc Netowrks," 15th International Conference on Advanced Communication Technology (ICACT’13), pp. 245-250, PyeongChang, Korea, January 2013.

[5] H. Han, M. Wu, Q. Hu, and N. Wang, "Best Route, Error Broadcast: A Content-Centric Forwarding Protocol for MANETs," IEEE 80th Vehicular Technology Conference (VTC Fall), pp. 14-17, Vancouver, Canada, September 2014.

[6] Y.-T. Yu, B.R. Dilmaghani, S. Calo, M.Y. Sanadidi, and M. Gerla, “Interest propagation in named data manets," in Proc. of IEEE International Conference on Computing, Networking, and Communications (ICNC’2013), pp. 1118-1122, San Diego, CA, January 2013.

[7] D. Kim and Y.-B. Ko, “A novel message broadcasting strategy for reliable content retrieval in multi-hop wireless content centric networks,” In Proceedings of the 9th ACM International Conference on Ubiquitous Information Management and Communication, Bali, Indonesia, January 2015.

[8] R.A. Rehman and B.-S. Kim, “Energy aware forwarding in content centric based multihop wireless ad hoc networks,” IEICE transactions on on Fundamentals of Electronics, Communications and Computer Sciences,vol. E98-A, no. 12, pp. 2738-2742, December 2015.

[9] M. Amadeo, C. Campolo, and A. Molinaro, “Forwarding strategies in named data wireless ad hoc networks: Design and evaluation,” Journal of Network and Computer Applications, vol. 50, pp. 148-158, April 2015.

[10] D. Kim, J.-H. Kim, C. Moon, J. Choi, and I. Yeom, “Efficient content delivery in mobile ad-hoc networks using ccn,” Ad Hoc Networks, vol. 36, pp. 81-99, January 2016.

[11] A. Afanasyev, I. Moiseenko, and L. Zhang, “ndnSIM: NDN simulator for NS-3,” Technical Report NDN-0005, NDN, October 2012.

INTENTIONAL BLANK

PROJECTION PROFILE BASED NUMBER PLATE LOCALIZATION AND RECOGNITION

Sandipan Chowdhury[1], Arindam Das[2], and Punitha P[2]

[1]Department of Computer Science and Engineering,Technology Campus
University of Calcutta, India
[2]Imaging Tech Lab, HCL Technologies, India
sandipanchowdhury6@gmail.com
arindam.d@hcl.com;drpunitha.S@hcl.com;

ABSTRACT

This paper proposes algorithms to localize vehicle number plates from natural background images, to segment the characters from the localized number plates and to recognize the segmented characters. The reported system is tested on a dataset of 560 sample images captured with different background under various illuminations. The performance accuracy of the proposed system has been calculated at each stage, which is 97.1%, 95.4% and 95.72% for localisation & extraction, character segmentation and character recognition respectively. The proposed method is also capable of localising and recognising multiple number plates in images.

KEYWORDS

Number plate localization, number Plate recognition, Character segmentation and recognition

1. INTRODUCTION

With overwhelming development of various computer vision techniques, a good number of applications have been employed in transport systems for varied purposes such as automatic toll tax collection, adding security measures in restricted areas, tracking in-out time of cars in parking lot and many more. Number plate recognition aiding the above purposes, also has a wide application in law enforcement and surveillance at traffic signals, speed limit junctions etc. Computer vision driven automatic number plate recognition systems are expected to automatically take photographs of vehicles and uniquely identify vehicles when they pass through certain points where these monitoring devices are installed.

Research findings in this domain can be dated back to 1970s, which is the result of the need to implement law enforcement and traffic control on transportation systems. The basic requirements of complete and accurate automation of this system, still keeps the research avenues in this area open even today. The ideal expectation for practical implementation and real-time usage is that the accuracy must be 100% and the computation complexity should be the possible minimum. A

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 185–200, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60615

few reasons which makes this a challenging task are the problems arising from natural/dynamic scene analysis; varying light conditions; different weather conditions; camera capturing limitations; Effects of distortion, blur and occlusion; language/scripts used; and those challenges raising with vehicle motions. In addition to these, some problems which cannot be addressed by computer vision solutions are worn out number plates which are not illuminated well during hours of darkness, dirty and broken etc. Most approaches therefore work only under restricted conditions such as fixed illumination, limited vehicle speed, designated routes, and stationary backgrounds.

This paper presents an algorithm to recognize number plates from natural scenes. The proposed method comprises of three main stages, localization of the number plate, character segmentation, and character recognition. Localisation stage includes binarization and noise elimination; locating the number plate region; extracting the number plates. An adaptive threshold based global binarization and locally applied Otsu's binarization are combined to obtain a more accurate binarization which retains the number plate and its characters intact. Localization & extraction of the number plate follows projection profile based approach. This approach helps to identify the possible regions of number plates in the images. These regions are further examined to select number plate region for further processing or to discard. This technique helps to localize and extract multiple number plates present in a single image.

In character segmentation, projection profile technique, in addition to an approximation algorithm is used to segment each character from the number plate. For character recognition, Support Vector Machine (SVM) is used. The segmented characters are classified into 36 classes, 26 classes of alphabets and 10 classes of numerals. The performance of the system has been computed at each stage and is found to be around 97.1%, 95.4% and 95.72%. The overall accuracy of the system is 92.68%.

The remaining of the paper is organized as follows. A brief background of related work is given in Section II. The proposed algorithm is described in Section III. Details of database are given in Section IV, Experimental results and comparison study is presented in Section V. Discussion and conclusions are drawn in Section VI.

2. RELATED WORK

Salter in 1984 [1], presented the potential applications of automatic vehicle identification for vehicle weighing and classification. Dickinson and Waterfall [2], in the same conference presented a general discussion on the suitability of image/video processing to perform collection of data, automatic surveillance, automatic incident detection, vehicle tracking, and vehicle classification. Since, then many researchers [1] - [26] have worked towards addressing various challenges in Number plate recognition.

Anagnostopoulos et al. [3], gives a survey on the work carried out till 2007. The article categorizes and assesses numerous techniques developed for license plate recognition in still images or video sequences. Shan Du et al., [7], present a comprehensive review of the state-of-the-art techniques for Automatic License Plate Recognition (ALPR) till 2013. The methods are categorized into different ALPR techniques according to the features the methods used at each stage, and compare them in terms of pros, cons, recognition accuracy, and processing speed.

Suresh et al., [4], proposed a novel method to enhance license plate numbers of moving vehicles in real traffic videos. Clemens et al., [5] present full-featured license plate detection and recognition system implemented on an embedded DSP platform processing video streams in real-time. Shapiro et al., [6], proposed an inexpensive automatic solution for remote vehicle identification. The recognition scheme combined adaptive iterative threshold with a template-matching algorithm. In [8], Chang et al., has attempted to take care of some restricted working environment conditions. The technique use fuzzy disciplines to extract license plates from an input image, and neural network aims to identify the number present in a license plate.

Azad and Shayegh [9], use adaptive threshold to obtain binary images and then use edge detection and morphological operations to localize number plates. Hsu et al., [10] proposed edge clustering mechanism for number plate detection, maximally stable extreme region (MSER) detector for character segmentation and LDA based character recognition. Chaudhary and Chincore [11] proposed 8-neighbor connectivity analysis to localize multiple number plates in Indian road conditions. Li et al., [12] used MSER to detect candidate characters and designed conditional random field models and through belief propagation inference estimated the license plate location. Carballido et al., [13] proposed a template matching approach to recognize license plate digits in outdoor parking entrance vehicles. Wen et al [15] proposed a method for number plate recognition which uses Bernsen's algorithm to binaries the images and connected component analysis to localize number plate regions and finally SVM to recognize the characters. Comeli et al., [16] have proposed a complete solution to recognize number plates by localizing the number plate through maximum local contrast, enhancing the images through Gaussian filters and histogram stretching, detecting and correcting tilt and finally recognizing characters through template matching. Many researchers have worked on number plate recognition of multi-style and multi-nations. Shiying et al [14], proposed a decision tree based localization of multinational license plate. Jiao et al., [20] proposed a morphology driven method for multi-style, multinational license plate recognition. Thome et al., [21], used gradient density to localize number plates and used hierarchical neural network to perform character recognition through classification. Guo and Liu [22], proposed a self-learning and hybridized technique for license plate localization. Al-Ghaili et al., [23] proposed license plate detection through the study of vertical line structures in the image. Naito et al., [19] proposed a robust license plate recognition system capable of recognizing number plates of moving vehicles in outdoors using wide dynamic range cameras. Chang [17] also proposed a line detection and projection based method to localize number plates and normal factors to recognize characters. Wang and Liu [18] used morphological operators and connected component analysis to localize the number plates. Poon et al., [25] also used several greyscale morphological operators to localize number plates. Sirithinaphong and Chamnongthai [24], used projection profiles to localize number plates and then used back propagation neural network to classify the characters.

The existing license plate recognition systems address the problem of recognition through three sub-tasks viz., license plate localization, character segmentation and character recognition. Similarly, in the next section we present in detail the algorithm we propose to recognize Indian number plates in outdoor scenes.

3. PROPOSED SYSTEM

The proposed system comprises three main stages, localization, character segmentation & normalization, and recognition.

3.1. Number Plate Localisation

The localisation stage focuses on detecting the number plate region from the captured image, and comprises of sub-stages, binarization, Noise removal through filtering, and region cropping.

3.2. Binarization

Let I be an RGB image (Fig-1).I is converted into 8-bit gray scale image (Fig-2) G by using the standard weights set by NTSC (National Television System Committee) which is given below.

$$G(u,v) = 0.229R + 0.587G + 0.114B \quad (1)$$

Figure 1. Input RGB image I

Figure 2. Gray scale image G of Fig.1

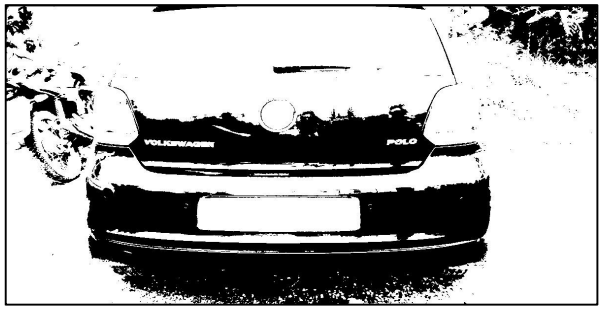


Figure 1. Binarized image B, using Otsu Global thresholding

The gray scale image G is then converted into binary image$B$, using well known Otsu's global threshold method (Fig-3). It was observed that while performing Otsu's method, although the image is pretty noiseless, the number plate gets eliminated due to global threshold. If the threshold is performed on local regions, the process can help to retain the number plate. To perform local threshold, G is divided into blocks of size 50×50 and Otsu's binarization is performed on each window of 50×50. Likewise window is shifted to acquire next 50×50 region of G to perform the binarization iteratively on all regions. The output of applying Ostu'sbinarization on the image by considering local regions of 50x50 is shown in Fig-4. This local thresholding will introduce some noise along with retaining the number plate.

Global thresholding provided comparatively noiseless image with no or some part of number plate whereas local thresholding helped us to get the number plate but with more noise.

Figure 4. Binarized image C, using Otsu's method on local regions

3.3. Connected Component Based Filtering

The resultant image produced in binarization stage is noisy in nature. To eliminate the unwanted noise we perform a connected component based filtering. The connected components are examined if they are part of the number plate or not. Based on it verification the component is retained or removed.

It has been observed from the number plates that after binarization, the characters of the number plate are surrounded by good amount of white pixel region. This is one of the characteristics of number plates, having dark characters over lighter backgrounds. The rectangular structure of the plate contains darker characters over the lighter background which is a mandate. The characteristic feature of having white pixels around the characters of the number plate that can be seen in C, is used to localize the number plate. Fig-5 shows an example of 8-neighbor connected component.

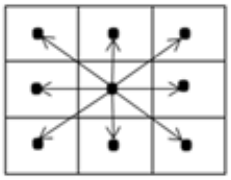


Figure 5. 8-Neighbor connectivity

To qualify a connected component as a possible number plate character we perform the following.

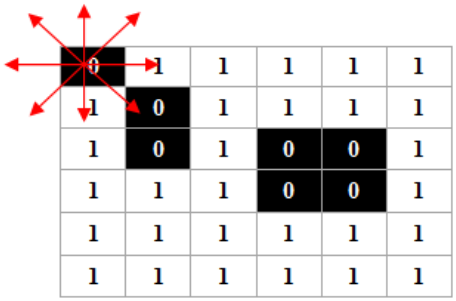
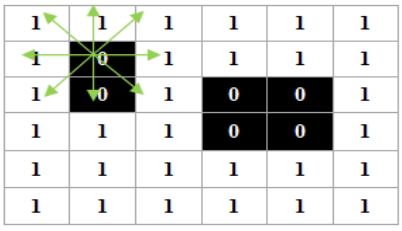


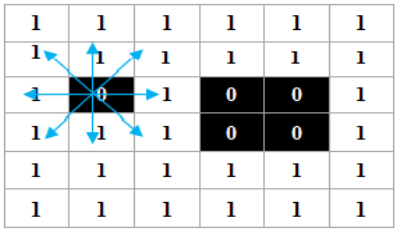
Figure 2. 8 neighbour connected component analysis starting from the border pixel

The image C is scanned at the borders for a black pixel. Once a black pixel if found, this pixel is used as a seed to find all its 8 connected neighbours, i.e., while scanning the image if any border pixel B_p, is found to be black then we place a3 × 3 window in such a way so that the pixel B_p becomes the center of the 3 × 3 window. Let this B_p be referred as the origin border pixel. Then the 8-neighbors which are also black and are connected to B_p are noted. Any one of the neighbours which is connected to B_p and is black in color becomes next center. The procedure is recursively repeated over all black pixels which are connected to B_p. All these pixels which form a region connected to the origin border pixel are eliminated, by changing them from black to white. For illustration we consider the below matrix, shown in Fig.6, where 0 represents the black pixel as the foreground and 1 represents the white pixel as the background.

In the above figure Fig-8 it is clearly observed that pixels (0,0), (1,1), (2,1), (2,3), (2,4), (3,3) and (3,4) are black pixels and among these pixels only (0,0) is the border pixel. So pixel (0,0) is made seed pixel. Now the 8 neighbour pixels of (0,0) which are also black are found. The pixel at (1,1) is found to be black, and again 8 connected neighbour analysis is done for (1,1). The process recursively continues till no more new connections are found. For the border pixel (0,0), (1,1) and (2,1) are found as the 8-connected neighbours. This region of three pixels is eliminated by replacing them by white pixels. This sequence is pictorially illustrated in Fig. 7.



(a)



(b)

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	0	0	1
1	1	1	0	0	1
1	1	1	1	1	1
1	1	1	1	1	1

(c)

Figure 7. 8-neighbor connected component analysis

The effect of this connected component based filtering, over a small region is shown in Fig-8.

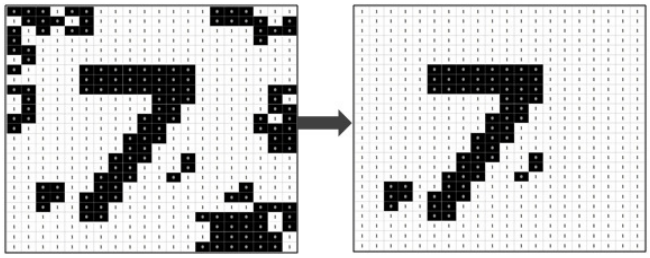


Figure 8. Effect of 8-neighbor connected component analysis

This 8 neighbour connected component analysis based filtering helps in getting a significantly noiseless image D(Fig-9).

Figure 9. Connected Component based filtered image D of input image C

3.4. Horizontal Black Run-length based Image Filtering

Though connected component analysis based filter removes significant amount of noise, some noises which are disconnected from the image border are generally not removed. Connected component analysis based filters starts from the border pixels and eliminates all pixels connected to the border pixels. This leaves some connected components which are present towards the center of the image which are disconnected from border pixels to be retained although they do not pertain to be number plate components. To remove such noises yet another filtering approach is proposed which performs at each row level.

Figure 10. Horizontal Run-length based Image Filtering E of input image D

Each row is scanned for continuous connected black pixels. A single row may have more than one black connected component. If the pixel count of these black run-length components counts is greater than 3% of the image width (V) then corresponding black pixels are converted to white. Also again we check the total black pixel count for each row, if the count gets less than the 5% or greater than the 80% of the image width of E then the entire row is made white.

A survey was carried out within our dataset and it is found that the width of any character present in the number plate is less than the 3% of the width of the image for resolution 1456x2592. This hypothesis helped us to get images with significantly less noise. On the other hand, if we observe that a row contains black pixels less than 5% or greater than 80% of the width of an image then we conclude that either the row doesn't have any part of the number plate or the row is full of noise. The effect of this image filtering technique has been furnished in Fig-10.

3.5. Number plate region extraction

While analyzing the pre-processed image E, it was clearly observed that the image can be segmented into a number of sub-images. The sub-images can be further analyzed to check the presence of the number plate. Along with number plate, sub-images may contain some additional noise components. Horizontal projection profile is used to segment the entire image into sub-images for further analysis. The horizontal projection profile for image E is shown in Fig-10.

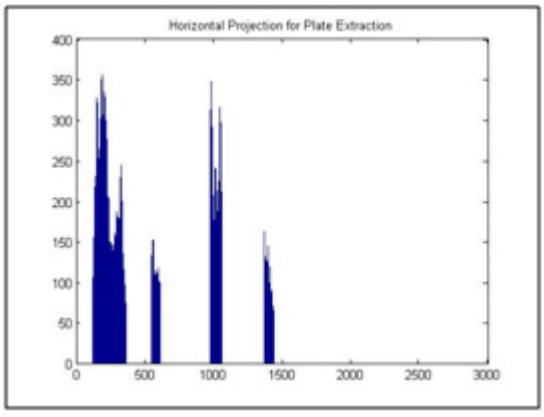


Figure 11. Horizontal Black Profile Projection of E

From the plot shown in Fig.11, if we find out the drastic shifts from zero and a drastic drop to zero, with the rise and a drop forming one region, in the above profile, we can see four sub-images. The number plate can be present in any of these four regions. Sometime, if the number plate runs over two lines, then part of the number plate can be present in two separate sub-images, which are adjacent to each other. The other sub-images can be discarded as irrelevant. We have to crop these regions from the image E by inspecting the projection profile. From our experimental observation, sub-images that correspond to noise or irrelevant regions tend to have lesser black pixels density than sub-images that correspond(s) to number plate region.

First we calculate the average black pixel density of the image E using equation (1).

$$M = \frac{Number\ of\ black\ pixels\ in\ E}{Size\ of\ E} \text{...(Eq.1)}$$

From the horizontal profile the maxima point of each dense segment is computed. If the maxima of any segment is higher or equal to M, i.e, the mean black pixel density of entire image, then the sub-image corresponding to that segment of the projection profile is considered to have number plate. However, we also need to ensure that the width of these sub-images is also acceptable. For this, we must calculate width of each segment in the projection profile. If the width of the segment is less than 2% of the length (Number of rows) of original image E then the sub-image corresponding to that segment of project profile can be discarded.

Figure 12. Extracted all probable regions of number plate

4. CHARACTER SEGMENTATION

From the above sub-images, to segment out the characters we perform two validating criteria explained below.

4.1. Projection Profile Based Validation

If the extracted sub-images are observed separately then it can be seen that the characters are separated by a very few white pixels and generally they maintain an equidistance from each other whereas the regions with full of noise do not have such observations. To achieve this, vertical black projection profile has been applied on extracted sub-images to determine the high density black region. One such projection profile is shown in Figure 13.

From the Fig-13, it is very clear that centre peak region represents high density black regions which can be regarded as characters of number plate, if we get black pixel peaks in the alternate segments separated by equidistant segments corresponding to background, until a long stretch of drop to zero is encountered.

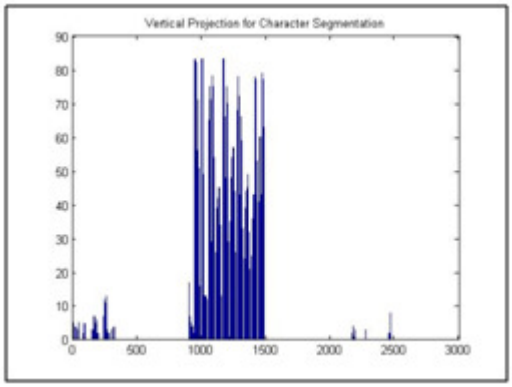


Figure 13. Vertical Black Projection Profile of a sub-image consisting of Number Plate

4.2. Character Height and Width Validation

As per our survey on our dataset, it has been observed that if the distance between camera and the car is nearly 10 meters and the resolution of the captured image is 1456x2592 then on an average a valid character on the number plate must contain at least 100 black pixels. Each sub-image is examined separately to discard connected components with size less than 100 pixels. Some of the outcomes of eliminating connected components of size less than 100 pixels for some sub-images are shown in Fig.14.

In addition to the above observation, the size of the characters of the number plate is also uniform in nature. The individual height and width of those character segments will be approximately same. After, eliminating some unwanted smaller components, a minimum bounding rectangle is fit over all remaining connected components.

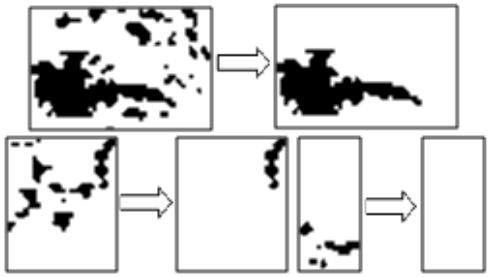


Figure 14. Removal of connected components of smaller size

The minimum bounding rectangles fit around the connected components corresponding to the characters belonging to number plate region, will approximately have the same height and width. It can also be observed that the, begin and end of these rectangles will be aligned to the same point on the Y-axis. So if there are at least 4 or more rectangles which are adjacent to each other and have the properties explained above, then these rectangles can be selected as relevant. We choose 4 rectangles due to the reason; the number plate characters may be spread in 2 lines, with a maximum of four characters in each line. Another criteria, that is employed to check if the segmented characters belong to the number plate is by comparing if the region of these characters match with the region which correspond to the dense projection profile generated in the previous stage.

Using the above two validation criteria, the character and the number plate regions are segmented and separated from the other irrelevant components as shown in Fig. 15. This segmented region along with the segmented characters is then passed to the character recognition stage.

Figure 15. Segments after Employing Height-Width Approximation method

(a)

(b)

Figure 16. Segments after Employing White Pixel Ratio over Black Pixel

In order to make sure that the classification stage receives only characters (Alphabets and numerals as inputs), white to black pixel ratio for each connected component enclosed in the minimum bounding rectangle is calculated. If the character enclosed in the rectangle has a ratio greater than 4 (refers to segment maximum white) or equal to 0 (refers to segment full black) then the segment is discarded. This technique produces noise free character segments only. An example of such case and the result after the elimination of the noisy segments is shown in Fig.16 (a) and (b). The non-alphabetical characters enclosed in minimum bounding rectangles are just simulated noise. If the total number of characters segmented is less than 4, it is discarded from next stage of recognition and classification. The segmented characters are normalized to a new scale of dimension (40×20) before sent for training and classification.

5. OPTICAL CHARACTER RECOGNITION

In this stage, Support Vector Machine (SVM) was used for supervised learning of 36 classes (26 alphabets of upper case and 10 digits).During dataset generation, it was observed that number plate generally contains alphabets in upper case only so the same data for lower case alphabets are not considered.

For character recognition, the entire character enclosed by a minimum bounding rectangle is used for training. The image is vectorised by re-arranging the pixel values into one dimensional vector K of size,1×800. Let F represents total number of vectors. These FxK feature vectors are submitted to SVM for training. The segmented characters from the number plate are then classified using the trained classifier, corresponding ASCII values are written in a text file to achieve complete OCR.

6. DATABASE

We have tested our proposed algorithm on 560 different images of license plates. As earlier mentioned, all images are captured at a distance nearly 10meters from the vehicle. The resolution

of all captured images is 1456x2592 and they are saved in standard JPEG format. Apart from this database, to establish the proposed method as resolution invariant we have scaled down and scaled up the images in different resolutions and performed the same operation. Detailed discussion is given in the next section.

7. EXPERIMENTAL RESULTS AND COMPARISON STUDY

To test the resolution invariant feature 100 images have been scaled down and scaled up in different resolution and employed with the proposed method. It has been observed that the performance of proposed method was accurate in extracting number plates and character segmentation for various sizes of images shown in Table-1.

Various image scales
508×904; 564×1004; 627×1116; 697×1240; 774×1378; 860×1531; 955×1701; 1061×1890; 1179×2100; 1310×2333; 1602×2851; 1762×3136; 1938×3450; 2132×3795; 2345×4175

Table1: Different image sizes on which the number plate extraction and character segmentation was tested.

The overall performance of the proposed method compared to other methods with respect to number plate extraction, character recognition is shown in Table-2, and Table-3. The accuracy of character segmentation is 95.4%.

It is to be mentioned that the proposed approach is capable of recognizing double lined number plate. An example of such example has been shown in Fig-21.

As the number plate extraction algorithm extracts all the probable number plate regions, so it is possible to recognize multiple number plates from a single image or a double line number plate. An Example of such case is given below in Fig-22 using a synthetic dataset.

Table 2: Performance comparison of the proposed system for number plate extraction

Methods	Number Plate Extraction (%)
Lee et al. [26]	94.4
Chiou et al. [27]	96.2
Shi et al. [28]	96.5
Wang et al. [29]	98
Chang et al. [30]	98
Deb et al. [31]	92.4
Jia et al. [32]	95.6
Kim et al. [33]	93.5
Duan et al. [34]	93.6
Roy et al. [35]	91.59
Proposed	97.1

Table 3: Performance comparison of the proposed system for number plate recognition

Methods	OCR Rate (%)
Lee et al. [26]	95.7
Shi et al. [28]	89.1
Chang et al. [30]	94.2
Proposed	95.72

8. DISCUSSION AND CONCLUSION

The above proposed system performs efficiently for wide variations in illumination conditions and different types of number plates. It has the features like double line and multiple number plate recognition. Though there are certain restrictions in this system like- different font style (e.g. italics) and colors of the number plate, excessive skewed number plate, which we consider for our future work.

Figure 17. Successful Extraction of Double Lined Number Plate

Figure 18. Detection of Multiple Number Plates from Single Input Image

REFERENCES

[1] D.R. Salter, 1984, The potential of automatic vehicle identification, International conference on road traffic data collection, Institution of Electrical Engineers, London, England.

[2] K.W.Dickinson, R.C. Waterfall, 1984, Video image processing for monitoring road traffic, International conference on road traffic data collection, Institution of Electrical Engineers, London, England.

[3] I.E.Anagnostopoulos, I.D.Psoroulas,V.Loumos,E. Kayafas, 2008, License plate recognition from still images and video sequences: a survey, IEEE Transactions on Intelligent Transportation Systems, Vol.9, Issue. 3,pp377 – 391.

[4] K.V. Suresh, G. Mahesh Kumar and A.N. Rajagopalan, 2007, Super-resolution of license plates in real traffic videos, IEEE Transactions on Intelligent Transportation Systems, Vol. 8, Issue. 2, pp321 -331.

[5] A.Clemens, F. Limberger,H. Bischof, 2007, Real-time license plate recognition on an embedded dsp-platform, IEEE Transaction on Intelligent Transportation Systems, vol. 1, Issue. 4, pp 34-54.

[6] V. Shapiro, G. Gluhchev, D. Dimov, 2006, Towards a multinational car license plate recognition system, Machine Vision and Applications. Volume 17, Issue 3, pp 173-183.

[7] S. Du, M. Ibrahim, M. Shehata, W Badawy,2013, Automatic license plate recognition (ALPR): a state-of-the-art review, IEEE Transactions on Circuits and Systems for Video Technology, Volume:23, Issue: 2, pp 311-325.

[8] S. Chang , L. Chen , Y. Chung and S. Chen, 2004, Automatic license plate recognition, IEEE Transactions on Intelligent Transportation Systems, Volume. 5, Issue. 1, pp 42 -53.

[9] R. Azad, H. R Shayegh, 2013, New method for optimization of license plate recognition system with use of edge detection and connected component, Third International Conference on Computer and Knowledge Engineering (ICCKE), pp 21 – 25.

[10] G.-S. Hsu, J.-C. Chen, and Y.-Z. Chung, 2013, Application-oriented license plate recognition, IEEE Transactions on Vehicular Technology, Volume. 62, Issue. 2, pp. 552–561.

[11] M.D. Chaudhary, J.B. Chinchore, 2014, Towards multiple license plate localization in Indian conditions: an edge density based approach, International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), pp. 60 – 65.

[12] B. Li, B. Tian, Y. Li, D. Wen, 2013, Component-based license plate detection using conditional random field model, IEEE Transactions on Intelligent Transportation Systems, Volume. 14, Issue. 4, pp. 1690-1699.

[13] J. Martínez-Carballido, R. Alfonso-López, J.M. Ramírez-Cortés, 2011, License plate digit recognition using 7×5 binary templates at an outdoor parking lot entrance, 21st International Conference on Electrical Communications and Computers (CONIELECOMP), pp. 18 - 21.

[14] M. Shiying, Z. Weixin, L. Na'na, S.Yaxin, H. Wen, 2011, Algorithm for multinational license plate localization and character segmentation, 10th International Conference on Electronic Measurement & Instruments (ICEMI) (Volume:3), pp. 85 – 89.

[15] Y. Wen, Y. Lu, J. Yan, Z. Zhou, K.M. Von-Deneen, P. Shi, 2011, An algorithm for license plate recognition applied to intelligent transportation system, IEEE Transactions on Intelligent Transportation Systems, Volume. 12, Issue. 3, pp. 830 - 845.

[16] P. Comelli, P. Ferragina, M.N. Granieri, F. Stabile, 1995, Optical recognition of motor vehicle license plates, IEEE Transactions on Vehicular Technology, Volume. 44. Issue. 4, pp. 790 – 799.

[17] J.K. Chang, 2010, Real-time vehicle license plate recognition on road images from various cameras,Third International Conference on Human-Centric Computing (HumanCom), pp. 1- 6.

[18] A. Wang, X. Liu, 2012, Vehicle license plate location based on improved roberts operator and mathematical morphology, Second International Conference on Instrumentation & Measurement, Computer, Communication and Control, pp. 995 – 998.

[19] T Naito, T Tsukada, K Yamada, K Kozuka, S Yamamoto, 2000, Robust license-plate recognition method for passing vehicles under outside environment, IEEETransactions on Vehicular Technology, Volume. 49, Issue. 6, pp. 2309 – 2319.

[20] J. Jiao, Q. Ye, Q. Huang, 2009, A configurable method for multi-style license plate recognition, Pattern Recognition 42, pp. 358-369.

[21] N Thome, A. Vacavant, L. Robinault, S. Miguet, 2011, A cognitive and video-based approach for multinational license plate recognition, Machine Vision and Applications, Volume 22, pp 389–407.

[22] J.M Guo, Y.F. Liu, 2008, License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques, IEEE Transactions on Vehicular Technology, Volume. 57, Issue. 3, pp. 1417 – 1424.

[23] A.M. Al-Ghaili, S. Mashohor, A.R. Ramli, A. Ismail, 2013, Vertical-edge-based car-license-plate detection method, IEEE Transactions on Vehicular Technology, Volume. 62, Issue. 1, pp. 26 – 38.

[24] T. Sirithinaphong, K. Chamnongthai, 1999,Recognition of car license plate for automatic parking system, Fifth International Symposium on Signal Processing and its Applications (ISSPA), pp. 455 – 457.

[25] J.C.H. Poon, M. Ghadiali, G.M.T. Man, L.M. Sheunp, 1995, A robust vision system for vehicle license plate recognition using grey-scale morphology,Proceedings of the IEEE International Symposium on Industrial Electronics, ISIE '95.

[26] H.-J. Lee, S.-Y. Chen, and S.-Z. Wang, Extraction and recognition of license plates of motorcycles and vehicles on highways, in Proc. Int. Conf. Pattern Recognition, 2004, pp. 356–359.

[27] Y.-C. Chiou, L. W. Lan, C.-M. Tseng, and C.-C. Fan, Optimal locations of license plate recognition to enhance the origin-destination matrix estimation, in Proc. Eastern Asia Soc. Transp. Stu., vol. 8. 2011, pp. 1–14.

[28] X. Shi, W. Zhao, and Y. Shen, Automatic license plate recognition system based on color image processing, Lecture Notes Comput. Sci.,vol. 3483, pp. 1159–1168, 2005.

[29] S. Z. Wang and H. J. Lee, A cascade framework for a real-time statistical plate recognition system, IEEE Trans. Inform. Forensics Security, vol. 2, no. 2, pp. 267–282, Jun. 2007.

[30] S.-L. Chang, L.-S. Chen, Y.-C. Chung, and S.-W. Chen, Automatic license plate recognition, IEEE Trans. Intell. Transp. Syst., vol. 5, no. 1, pp. 42–53, Mar. 2004.

[31] K. Deb, H.-U. Chae, and K.-H. Jo, Vehicle license plate detection method based on sliding concentric windows and histogram, J.Comput., vol. 4, no. 8, pp. 771–777, 2009.

[32] W. Jia, H. Zhang, X. He, and M. Piccardi, Mean shift for accurate license plate localization, in Proc. IEEE Conf. Intell. Transp. Syst., Sep. 2005, pp. 566–571.

[33] S. K. Kim, D. W. Kim, and H. J. Kim, A recognition of vehicle license plate using a genetic algorithm based segmentation, in Proc. Int. Conf. Image Process., vol. 2. 1996, pp. 661–664.

[34] T. D. Duan, T. L. H. Du, T. V. Phuoc, and N. V. Hoang, Building an automatic vehicle license-plate recognition system, in Proc. Int. Conf. Comput. Sci. RIVF, 2005, pp. 59–63.

[35] A. Roy, D. P. Ghosal, Number Plate Recognition for Use in Different Countries Using an Improved Segmentation, in Proc. 2nd National Conf. on Emerging Trends and Applications in Computer Science, March 2011, pp. 1-5.

A TEXT MINING RESEARCH BASED ON LDA TOPIC MODELLING

Zhou Tong[1] and Haiyi Zhang[2]

Jodrey School of Computer Science, Acadia University, Wolfville, NS, Canada
[1]zhoutong@acadiau.ca
[2]haiyi.zhang@acadiau.ca

ABSTRACT

A Large number of digital text information is generated every day. Effectively searching, managing and exploring the text data has become a main task. In this paper, we first represent an introduction to text mining and a probabilistic topic model Latent Dirichlet allocation. Then two experiments are proposed - Wikipedia articles and users' tweets topic modelling. The former one builds up a document topic model, aiming to a topic perspective solution on searching, exploring and recommending articles. The latter one sets up a user topic model, providing a full research and analysis over Twitter users' interest. The experiment process including data collecting, data pre-processing and model training is fully documented and commented. Further more, the conclusion and application of this paper could be a useful computation tool for social and business research.

KEYWORDS

topic model, LDA, text mining, probabilistic model

1. INTRODUCTION

As computers and Internet are widely used in almost every area, more and more information is digitized and stored online in the form of news, blogs, and social networks. Since the amount of the information is exploded to astronomical figures, searching and exploring the data has become the main problem. Our research is intended to design a new computational tool based on topic models using text mining techniques to organize, search and analyse the vast amounts of data, providing a better way understanding and mining the information.

2. BACKGROUND

2.1. Text Mining

Text mining is the process of deriving high-quality information from text [1]. Text mining usually involves the process of structuring the input text, finding patterns within the structured data, and finally evaluation and interpretation of the output. Typical text mining tasks include text categorization, text clustering, document summarization, keyword extraction and etc. In this

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 201–210, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60616

research, statistical and machine learning techniques will be used to mine meaningful information and explore data analysis.

2.2. Topic Modelling

In machine learning and natural language processing, topic models are generative models, which provide a probabilistic framework [2]. Topic modelling methods are generally used for automatically organizing, understanding, searching, and summarizing large electronic archives.

The "topics" signifies the hidden, to be estimated, variable relations that link words in a vocabulary and their occurrence in documents. A document is seen as a mixture of topics. Topic models discover the hidden themes through out the collection and annotate the documents according to those themes. Each word is seen as drawn from one of those topics. Finally, A document coverage distribution of topics is generated and it provides a new way to explore the data on the perspective of topics.

2.3. Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) is a generative model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar [3]. LDA has made a big impact in the fields of natural language processing and statistical machine learning and has quickly become one of the most popular probabilistic text modelling techniques in machine learning.

Intuitively in LDA, documents exhibit multiple topics [4]. In text pre-processing, we exclude punctuation and stop words (such as, "if", "the", or "on", which contain little topical content). Therefore, each document is regarded as a mixture of corpus-wide topics. A topic is a distribution over a fixed vocabulary. These topics are generated from the collection of documents [5]. For example, the sports topic has word "football", "hockey" with high probability and the computer topic has word "data", "network" with high probability. Then, a collection of documents has probability distribution over topics, where each word is regarded as drawn from one of those topics. With this document probability distribution over each topic, we will know how much each topic is involved in a document, meaning which topics a document is mainly talking about.

A graphical model for LDA is shown in Figure 1:

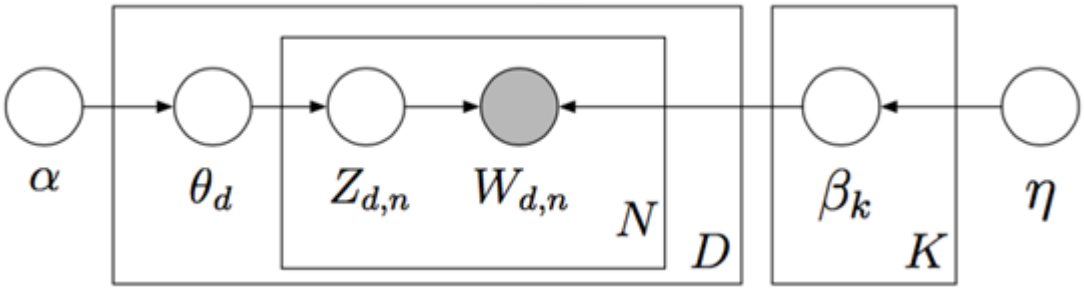


Figure 1. Graphic model for Latent Dirichlet allocation

As the figure illustrated, we can describe LDA more formally with the following notation. First, α and η are proportion parameter and topic parameter, respectively. The topics are $\beta_{1:K}$, where each β_k is a distribution over the vocabulary. The topic proportion for the d th document are θ_d , where $\theta_{d,k}$ is the topic proportion for topic k in document d . The topic assignments for the d th document are Z_d, where $Z_{d,n}$ is the topic assignment for the n th word in document d . Finally, the observed words for document d are w_d, where $w_{d,n}$ is the n th word in document d , which is an element from the fixed vocabulary.

With this notation, the generative process for LDA corresponds to the following joint distribution of the hidden and observed variables:

$$p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}) = \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d) (\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n}))$$

Notice that this distribution specifies a number of dependencies. The topic assignment $Z_{d,n}$ depends on the per-document topic distribution θ_d; and the word $w_{d,n}$ depends on all of the topics $\beta_{1:K}$ and the topic assignment $Z_{d,n}$.

2.4. Jensen-Shannon Divergence

In probability theory and statistics, the Jensen-Shannon divergence is a popular method of measuring the similarity between two probability distributions. It is also known as information radius or total divergence to the average. It is based on the Kullback-Leibler divergence. The square root of the Jensen-Shannon divergence is a metric often referred to as Jensen-Shannon distance [6].

For discrete probability distributions P and Q, Kullback-Leibler divergence of Q from P is defined to be:

$$D_{KL}(P \| Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

So, the Jensen-Shannon divergence of Q from P is defined by:

$$JSD(P \| Q) = \frac{1}{2} D_{KL}(P \| M) + \frac{1}{2} D_{KL}(Q \| M)$$

where $M = \frac{1}{2}(P + Q)$

Jensen-Shannon divergence measures the similarity between two distributions. By applying Jensen-Shannon divergence to the topic assignment for the d th document Z_d, it will allow us to measure the distance and similarity between each document.

3. DESIGNS AND EXPERIMENTS

In order to apply topic modelling and propose a new text mining solution on topics, we have designed two experiments fulfilling our goal. The first one is to use topic modelling manage and explore Wikipedia, and the second is a Twitter application on topic modelling. The former is a new solution on a typical problem and the latter is building up a new model on Twitter data analysis.

3.1. Wikipedia on Topic Modelling

3.1.1. Experiment Overview

Wikipedia is a free-access, free-content Internet encyclopaedia, supported by non-profit Wikimedia Foundation. It has millions of articles for people to search, explore or even edit. In this experiment, the text data is from simplified Wikipedia (English version) with over 200,000 articles. By applying Latent Dirichlet allocation (LDA) and topic modelling, a solution of topic searching, exploring and recommending system will be achieved.

3.1.2. Data Pre-processing

The simplified Wikipedia English version is free for download from Wikipedia Foundation database backup dumps. The backup is in a format of XML. The first step of data pre-processing is to parse the XML file and extract the text data. R package *XML* provides a series of function parsing XML file. By using those functions, we will get a relatively clear data of all the articles in a data frame.

The next step is text-cleaning process. The purpose of text cleaning is to simplify the text data, eliminating as much as possible language dependent factors. Articles are written in natural language for human to understand. But in text mining, those data are not always easy for computers to process. In this experiment, there are three steps in text cleaning:

- Tokenization: a document is treated as a string, removing all the punctuations and then partitioned into a list of tokens.

- Removing stop words: stop words such as "the", "if", "and" ... are frequently occurring but no significant meanings which need to be removed.

- Stemming word: stemming word that converts different word form into similar canonical form. For example, computing to compute, happiness to happy. This process reduces the data redundancy and simplifies the later computation [7].

3.1.3. Model Training

The training process requires R package *topicmodels* with its package dependencies (*tm* and others) to be loaded. An LDA model of simplified English Wikipedia on a sample of 1000 articles with more than 1000 characters, returned after 2000 iterations of Gibbs sampling, with $K = 50$ topics, and Dirichlet hyper-parameters $\beta = 0.1$ and $\alpha = 50/K$. Meanwhile, topic distribution coverage for each document is generated. This distribution represents how much each

document is related to each topic. A new way of search and explore documents over topics can be implemented.

Table 1. A few selected topics generated from Wikipedia topic distribution

Topic 2	Topic 13	Topic 22	Topic 31	Topic 46
athlete	album	universal	movie	hurricane
olympia	song	college	categories	major
field	music	school	fiction	season
summer	record	categories	solstice	minor
track	band	new	alien	key
men	release	economic	star	storm
metre	single	institut	sun	tropic
image	rock	educate	direct	chord
women	singer	science	drama	verse
medal	pop	work	southern	end

Table 1 shows five selected topic terms after the model is trained, where top ten terms are listed for each topic. With LDA training, the terms in the same topic tend to be similar. Formally speaking, they are highly associated. For example, topic 13 is about music, topic 22 is about education and topic 46 is about weather. This topic distribution provides a way to search topic and explore between topics in order to find the document the user is looking for.

After the model is built, Jensen-Shannon divergence is applied to calculate the similarity of each distribution. Sorting the similarity of one document between every other distribution, a topic recommender system can be implemented.

3.1.4. Results

Here is an example of article *Light* from the experiment. The original article is shown in the left part of Figure 2, which can be also accessed in simple Wikipedia online (the data for this experiment is retrieved as a backup version on 11/1/2016). After the model is trained, we got a series of article distribution over each topic. The right part of Figure 2 is the bar plot of article *light* topics distribution. In total of 50 topics, we can easily find there are 3 topics with obviously high probabilities – topic 47, topic 45 and topic 16. Table 2 shows the top 5 probabilities topics. These probabilities are how tightly this article is associated with each topic. Table 3 shows the terms in these 5 topics with 10 terms each.

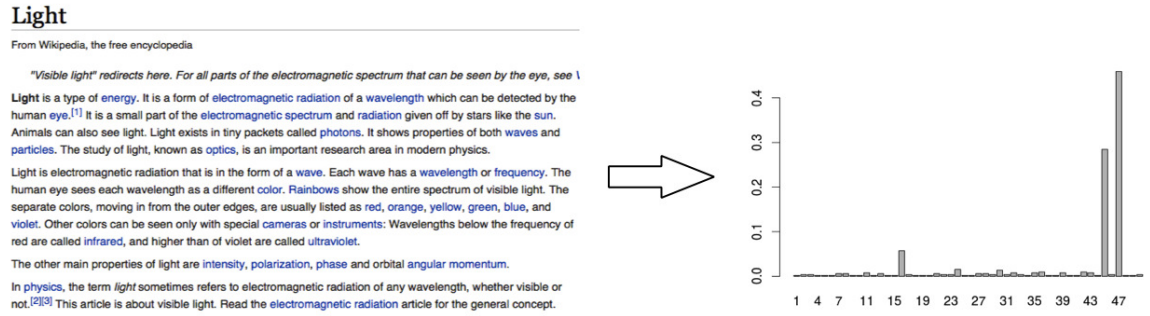


Figure 2. Topics Distribution of Article *Light*

Table 2. Top 5 Topic Probabilities of Article *Light*

Topics	Topic 47	Topic 45	Topic 16	Topic 24	Topic 30
Probabilities	0.05692600	0.45920304	0.28462998	0.01518027	0.01328273

Table 3. Top 5 Topic Terms of Article *Light*

Topic 47	**Topic 45**	**Topic 16**	**Topic 24**	**Topic 30**
light	use	color	computable	cleaner
beamline	one	style	equation	people
beam	also	background	fluid	use
radiated	can	hex	categories	clean
don	people	rgb	image	make
synchrotron	mania	magenta	program	chemical
wavelength	call	ffffff	protocol	thing
physical	time	fuchsia	mathematical	paint
station	two	red	function	made
carlo	like	pink	design	put

So just like this example, every article of the whole collection is represented as a vector of probabilities over 50 topics. This is the core data of our model, where we can do all sorts of applications. Here is an example of finding the most related article of article *Light*. We will use Jensen-Shannon divergence to calculate the distance between article *Light* and every other article. The shortest distance will be the most related article. After calculating the distance, Table 4 shows the top 10 of the shortest articles. To be noted, the distance should have been the square root of the distance below, but to simplify the calculation and higher the accuracy, we will stay with the squared number, as there would be no difference on comparing the closet distance.

Table 4. Top 10 Shortest Distance of Article *Light*

Articles	Article 855	Article 820	Article 837	Article 299	Article 911
Distance^2	0.0000000	0.2271741	0.3404084	0.3881467	0.4583745
Articles	Article 341	Article 287	Article 328	Article 544	Article 606
Distance^2	0.4671845	0.4728383	0.4802241	0.4803357	0.4874499

As shown on Table 4, these 10 articles are the closest distance with Article *Light*. The distance of Article 855 is 0, because it is article *Light* itself. So Article 820 is the closest distance with *Light*, meaning their contents most related. Meanwhile, what we get is a sorted list of closest distance and will also work if we require more than one most related article. This method is based on calculating the probabilities each meaningful word in the model. So the accuracy is much better than calculating the keywords, or titles, which is widely used on many documents management system. Lastly, if we look up the title of Article 820, it is *Beamline*. Based on the 2000 article dataset, it is a convincing answer to the most related article of *Light*.

3.2. Twitter Data Analysis

3.2.1. Experiment Overview

Twitter is an online social networking service that enables users to send and read short 140-character messages called "tweets". Currently, Twitter has more than 332 million active users posting 340 million tweets a day. Twitter has a big impact in everyday life. Twitter mining is not

only a big research task in computer science or statistics, but also a key factor on social and business research. In this experiment, by applying LDA and topic modelling, a deep research and analysis on Twitter users is proposed. A detail model of Twitter user's personality and preference will be inferred.

3.2.2. Data Pre-processing

Generally, a tweet from any twitter user is public and free to read for everyone. Having registered as a Twitter application developer will allow you to access all the tweets and a set of APIs to manipulate the data [8]. The first step is to collect tweets from the users. By using the APIs and R package *twitteR*, a sample of 10,000 valid users is gathered into a data frame. To improve the data quality, a standard is set up for the twitter user to be a "valid" user:

- The user profile is an unprotected, which means user's information and tweets is public to everyone. If the user sets the profile to be protected, his or her information cannot be gathered by developers.

- The user has at least 100 tweets. According to a statistical research from Twitter on January 2012, the average length of a tweet is 67.9 characters. Therefore, before pre-processing, there are 6790 characters for one user in a sample. Less than 100 tweets per user will lower the calculation quality.

- The user must use English as major language in the tweets. Some odd non-English word will not affect the model, but a number of total non-English users' data will longer the pre-processing time, add confusion to the topics, or even mess up the result.

The text cleaning process is basically the same as the Wikipedia experiment. Tokenization, stop words removing and word stemming is required in this process. However most tweets are oral and informal language, a few details need to be noticed:

- Some tweets may contain URLs, using hash tags on a topic, using @ to mention other users. In text cleaning process, these situations need particular functions to remove or parse.

- Some Internet terms, such as "LOL", "OMG", "BTW", are abbreviation of a phrase. Those terms can treat as stop words to delete.

- Some words are written as shorthand, such as "ppl" (people), "thx" (thanks), "fab" (fabulous). Those words need to stem to the original form.

3.2.3. Model Training

The training process is also similar to the Wikipedia experiment. However, Twitter data are formed with natural daily language, which has a narrow topic range compared to Wikipedia. The topics number is less and not with equally clear boundaries. Nevertheless, the topic model still has a good performance and the coverage distribution can easily illustrate the user's personal interest. Here is an experiment a sample of 100 twitter user with more than 100 tweets, returned after 2000 iterations of Gibbs sampling, with 30 topics.

Table 5. A few selected topics generated from Twitter topic distribution

Topic 1	Topic 5	Topic 9	Topic 17	Topic 28
halifax	check	stream	follow	weight
nova	reward	live	win	loss
scotia	one	league	enter	diet
man	kangaroo	communities	canada	news
refuge	new	check	retweet	lose
media	facebook	ea	card	natural
say	get	design	sunday	tip
woman	post	weekend	away	plan
central	photo	chat	chance	health
student	coffee	gold	donate	techno

Table 2 shows five selected topics terms after the Twitter topic model is trained. Similar to the previous experiment, the LDA model has a good performance on dividing topics. But as we expected, twitter data is limited by the daily language that leads to less clear boundaries as Wikipedia topics. For example, in topic 5, it is hard to label this topic into a particular category. However, the Twitter application is successful on building up topic models over users, and it will certainly benefit the statistic analysis and even a big impact on social and business research.

3.2.4. Results

This experiment has a similar structure with the Wikipedia articles. Instead of each article, we will treat every user's tweets as an article. With the topic model, we can also calculate the distribution over each topic. Here is an example of Tim Cook's Twitter. As shown in Figure 3, the left part is the Twitter user, and the right part is the bar plot of topic distribution. This distribution represents what kind of topic the user talks more and more interested. Table 6 shows the top 5 topics of Tim Cook talks most about. By applying Jenson-Shannon divergence to calculate the distance, we can find the people that talks the most similar topics or even with similar personality.

Figure 3. Topics Distribution of Twitter of Tim Cook

Table 6. Top 5 Topic Terms of Tim Cook's Tweets

Topic 19	Topic 12	Topic 20	Topic 15	Topic 22
day	new	apple	custom	summer
today	great	iphone	service	student
thank	canada	trump	expect	job
will	join	deal	job	american
great	congrat	say	product	wed
time	congratulations	app	donet	still
get	communities	product	photo	campus
ea	forward	vs	price	fair
can	proud	court	easier	act
new	event	camera	facebook	hall

4. CONCLUSIONS AND FUTURE WORK

In this paper, an introduction of text mining and topic model LDA is represented. We proposed two experiments, which built up topic models on Wikipedia articles and Twitter users' tweets. A brief introduction of each experiment including overview, pre-processing and model training is given and analysed.

With these data and model foundation, a number of future works can be done for further research and experiment.

- As the limitation of the computation power, this research is based on a relatively small sample by the time we start writing. However, the result is quite convincing even with the small size. Applying to a larger dataset will more likely achieve better results.

- An application on topic modelling to manage, search and explore offline Wikipedia articles could be implemented.

- A full research on Twitter users' interest could be applied. Further more, this application could be a useful tool for social and business research

- In Twitter application on topic modelling, we ignore the pictures users posted. What if we can combine image processing and topic model to provide a better performance?

REFERENCES

[1] Martin Ponweiser (2012) Latent Dirichlet Allocation in R, Vienna University of Business and Economics.

[2] Bettina Grun, kurt Hornik (2011) "topicmodels: An R Package for Fitting Topic Model", Journal of Statistical Software Vol. 40, No. 13.

[3] Qi Jing (2015) Searching for Economic Effects of User Specified Event Based on Topic Modelling and Event Reference, Jordery School of Computer Science, Acadia University.

[4] David M.Blei (2012) "Probabilistic Topic Models", Communications of the ACM Vol. 55, No. 4, pp77-84.

[5] David M.Blei, John D. Lafferty (2006) "A Correlated Topic Model of Science", Annals of Applied Statistics Vol. 1, No. 1, pp17-35.

[6] Jianhua Lin (1991) "Divergence Measures Based on the Shannon Entropy", IEEE Transactions on Information Theory Vol. 37, No. 1, pp145-151.

[7] Aurangzeb Khan, Baharum Baharudin, Lam Hong Lee, Khairullah khan (2010) "A Review of Machine Learning Algorithms for Text-Document Classification", Journal of Advances in Information Technology Vol. 1, No. 1, pp4-20.

[8] Yanchang Zhao (2015) R and Data Mining, http://www.rdatamining.com.

AUTHORS

Zhou Tong is currently a master student of computer science at Acadia University, Canada. His research is focusing on text mining.

Haiyi Zhang received his MS degree in 1990 from the Computer Science department of New Jersey Institute of Technology of USA, and his Ph.D in 1996 from Harbin Institute of Technology in China. He was a post-doctor in information department of ABO, Finland in 2000. His research interests are machine learning, data mining. He has more than 50 academic papers published. Currently he is an associate professor at Acadia University, Canada.

MANAGEMENT ARCHITECTURE FOR DYNAMIC FEDERATED IDENTITY MANAGEMENT

Daniela Pöhn[1] and Wolfgang Hommel[1]

[1]Leibniz Supercomputing Centre, Munich Network Management Team,
Garching n. Munich, Germany
poehn@lrz.de, hommel@lrz.de

ABSTRACT

We present the concept and design of Dynamic Automated Metadata Exchange (DAME) in Security Assertion Markup Language (SAML) based user authentication and authorization infrastructures. This approach solves the real-world limitations in scalability of pre-exchanged metadata in SAML-based federations and inter-federations. The user initiates the metadata exchange on demand, therefore reducing the size of the exchanged metadata compared to traditional metadata aggregation. In order to specify and discuss the necessary changes to identity federation architectures, we apply the Munich Network Management (MNM) service model to Federated Identity Management via a trusted third party (TTP); an overview of all components and interactions is created. Based on this model, the management architecture of the TTP with its basic management functionalities is designed. This management architecture includes further functionality for automated management of entities and dynamic federations.

KEYWORDS

Federated Identity Management, SAML, Service Management, Management Architecture, Trust Management

1. INTRODUCTION

Organizations, such as universities, provide several services to their members, e.g., email, exam management, and video conferencing. Users within the organization typically log in via username and password with optional additional factors, such as smartcards or X.509v3 user certificates. Their authorization is based on their roles and optionally manually assigned permissions, which are commonly stored in the organization's centralized Identity & Access Management (I&AM) system. The I&AM system is typically based on Lightweight Directory Access Protocol (LDAP) servers or relational database management systems in the backend. When users are part of a project, which is jointly carried out by several organizations, inter-organizational identity management becomes necessary. In order to allow users to re-use their home organization's accounts for external services, *Federated Identity Management (FIM)* was introduced. It facilitates the identity management between different organizations. While the OASIS standard Security Assertion Markup Language (SAML) [1] enables the exchange of user information in

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 211–226, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60617

trust boundaries, OpenID Connect uses the “trust and accept all comers” paradigm. Research & Education (R&E) and many industry sectors mainly depend on the trust model offered by SAML. SAML divides participating organizations into identity providers (IDPs), which are the home organizations of the users running an I&AM system, and service providers (SPs), which operate the services that are to be used in an inter-organizational manner. These entities, i.e., all IDPs and SPs, operate within trust boundaries that are called federations. The trust boundaries are specified by the SAML metadata of the involved entities, which contains information about the communications' endpoints, e.g., X.509v3 signature certificates, used SAML bindings, and URLs for connection establishment. While SAML does not force the pre-exchange of the aggregated, XML-based SAML metadata within geographic and industrial-sector-specific borders, this has become common practice, which means that there are many industry-specific and national federations. The limitation of SAML is, at the same time, that both the IDP and the SP need to possess each other’s SAML metadata before the user can login to the service and user profile information can be transferred from the IDP to the SP. As collaborations are not restricted to such artificial federation borders, *Inter-Federated Identity Management (IFIM)* was introduced. IFIM builds an umbrella federation over the existing federation by policies, contracts, and the pre-exchange of the aggregated metadata of all member federations. Although the additional contracts required between federations and their members make the inter-federation more complex and cumbersome to manage, the inter-federation eduGAIN [2] is significantly growing, already covering 40 national federations in the R&E environment. The growth amplifies another problem: the inter-federation metadata file is huge, making it cumbersome to process even on state-of-the-art hardware and slowing down the user experience.

Therefore, a more scalable approach for metadata exchange via a trusted third party (TTP) was introduced in the project GÉANT-TrustBroker (GNTB) [3]. In GNTB, the user initiates the metadata exchange during the first-time login to a specific service. For that reason, SPs and IDPs solely integrate the necessary metadata instead of the metadata of all IDPs and SPs within the federation. GNTB works as a TTP, which primarily is a central metadata repository. Alternatively, entities can register URLs pointing to their SAML metadata location, which often is publically accessible. In order to establish technically trusted SAML connections on-demand, the TTP extends the so-called SAML discovery service. The service, formally known as WAYF (Where Are You From?), is used to localize the user's IDP and therefore knows both endpoints of the metadata exchange.

The user wants to make use of a service, i.e., he expresses his will to access a specific service at a SP. By that, if IDP and SP technically do not know each other beforehand, the TTP triggers the metadata exchange on-demand. The involved entities can apply the Metadata Query Protocol [4] by Young for the actual metadata exchange, while the TTP orchestrates the overall exchange process. This workflow has been submitted for standardization by IETF in the Internet-Draft Dynamic Automated Metadata Exchange (DAME) [5]. The TTP is only involved during the first contact between IDP and SP and does not interfere in further communication. In order to integrate the metadata automatically, an extension of existing IDP and SP software packages is needed. This eliminates the manual workload for SP and IDP administrators and avoids waiting time for the end users. As only the necessary metadata is exchanged, this significantly improves the scalability of the metadata exchange, while at the same time avoids performance bottlenecks.

For example, the Leibniz Supercomputing Centre (LRZ) is part of the inter-federation eduGAIN, which currently includes about 1,030 SPs. This means that LRZ’s IDP has 1,030 potential trust

relationships with SPs plus the metadata of further 1,487 IDPs, which are not relevant for an IDP. In practice, however, only 4 SPs are used at the most, while the metadata of 2,517 entities are exchanged including the own metadata. To complicate it further, also the metadata of all entities of the German federation DFN-AAI are exchanged, which are not part of the inter-federation. With dynamic metadata exchange, the number of received metadata is reduced to 4.

GNTB is currently implemented and improved within the project GÉANT GN4, which operates the inter-federation eduGAIN. The implementation of the TTP is based on the open source SAML implementation Shibboleth. The Internet-Draft is advanced by the REFEDS community, which will establish a new working group for FIM at large scale at the IETF in 2016.

This paper focuses on the design of a service model and the management architecture for this TTP. A management architecture is a framework for management-relevant information, organization, communication, and functionalities. By implementing the management architecture, the TTP becomes a management platform. This management platform adds functionalities to the technical TTP, helping IDPs and SP establishing and managing trust relationships. While the GNTB implementation is tailored for SAML, the TTP and all its functionalities are generically designed, so it can be adopted to other FIM protocols, such as OpenID Connect, without changes. The service model for federated access management, described in Section 2, is based on the Munich Network Management (MNM) service model. The service model for FIM is applied on the TTP, explained in detail in Section 3. The different views provided by the service model help to establish a common understanding about service-related terms and to specify the service functionality additionally to the management tasks. The service model is the basis for the management architecture, described in Section 4. The management architecture describes the organizational, informational, communication, and functional model for a management platform. The management platform adds functionalities to the technical TTP, helping IDPs and SP managing trust relationships, and contributes to service management, which is discussed in Section 5. The paper is concluded by a summary and an outlook to our future work in Section 6.

2. SERVICE MODEL FOR FIM

The MNM service model [7][8][9] is a generic model for IT service management, defining service-related terms, concepts, and structuring rules. It allows to model specific services for the purpose to analyse needs and demands in regard to an appropriate service management with quality of service (QoS) guarantees. The MNM service model consists of three different partial and views: the *basic service model*, the *service view*, and the *realization view*. The basic service model contains the relevant roles and associations. It distinguishes between customer side, provider side, and side-independent aspects. The customer side consists of the basic roles *customer* and *user*, while the role *provider* is part of the provider side. The provider makes the service available to the customer side, whereas more details about the service are provided by the two views.

The service view focuses on the components between service provider and customer side, while the realization view is appropriate to identify objects within the provider side. The combined views provide a detailed service description. The service view, therefore, contains the functionality of the service, i.e., usage for the role *user* and management functionality, which is accessed by the role *customer*. The realization view, in contrast, describes the service implementation and the service management implementation. Both depend on provider-internal

resources (hardware, knowledge, and staff) and sub-services. While the service implementation serves to provide the service, the service management implementation includes a service management logic using basic management functionalities and external management sub-services.

As the MNM service model can be applied to arbitrary IT services, also recursively, we design the FIM service model on top of it. Based on the FIM service model, the dynamic automated metadata exchange via a TTP is designed in the service model. The detailed overview of all involved components can help to regard the security of a service and the interfaces. The basic service model for FIM contains the identified roles in the service interaction, as shown in Figure 1 (a). The roles are associated with different domains: the customer side, the service independent side, and the provider side. The notation of the roles is based on SAML, though the service model can be used for other protocols, like OpenID Connect. The customer side contains the IDP as well as the user, as the service is provided by the SP. Furthermore, the IDP can be a member of a federation, the so-called identity provider federation. The SP is part of the provider side, while the service provider can be a member of a federation, the so-called service provider federation. The service itself is independent of provider and customer side. A SAML attribute authority, which extends IDP functionality, can be placed on the side-independent part. This view can optionally be divided into different models as the IDP can be seen as a provider for the customer SP, which needs user information.

3. SERVICE MODEL FOR FIM WITH A TTP

In this section, we demonstrate the application of the MNM service model to FIM by modelling the dynamic metadata exchange via a TTP. This allows to identify the differences between FIM with pre-exchanged metadata and FIM with dynamic metadata exchange via a TTP. Furthermore, the service model explains the interactions between IDP, SP, and the user, while specifying the implementation from a service point of view. This will then become the basis for the management architecture.

In contrast to the basic service model for FIM, the service model for dynamic automated metadata exchange includes the side-independent TTP used to dynamically exchange the metadata. The TTP, as shown in Figure 1 (b), does not need to be part of a federation, though a federation, inter-federation, or any another trustworthy organization could operate it. As a result, the customer side consists of the user as well as the IDP and might include the identity provider federation. The provider side involves a service provider and an optional service provider federation. The side-independent part comprises the service, an optional attribute authority and the TTP.

After applying the basic service model to the dynamic automated metadata exchange via a TTP, the service view of the model is designed. This is shown in Figure 2. In order to better differentiate between service, SP, IDP, and the TTP, another part was added to the service view. The former customer side is renamed into IDP side, while the provider side is now called SP side. Along with the side-independent part a fourth side was introduced: the TTP side.

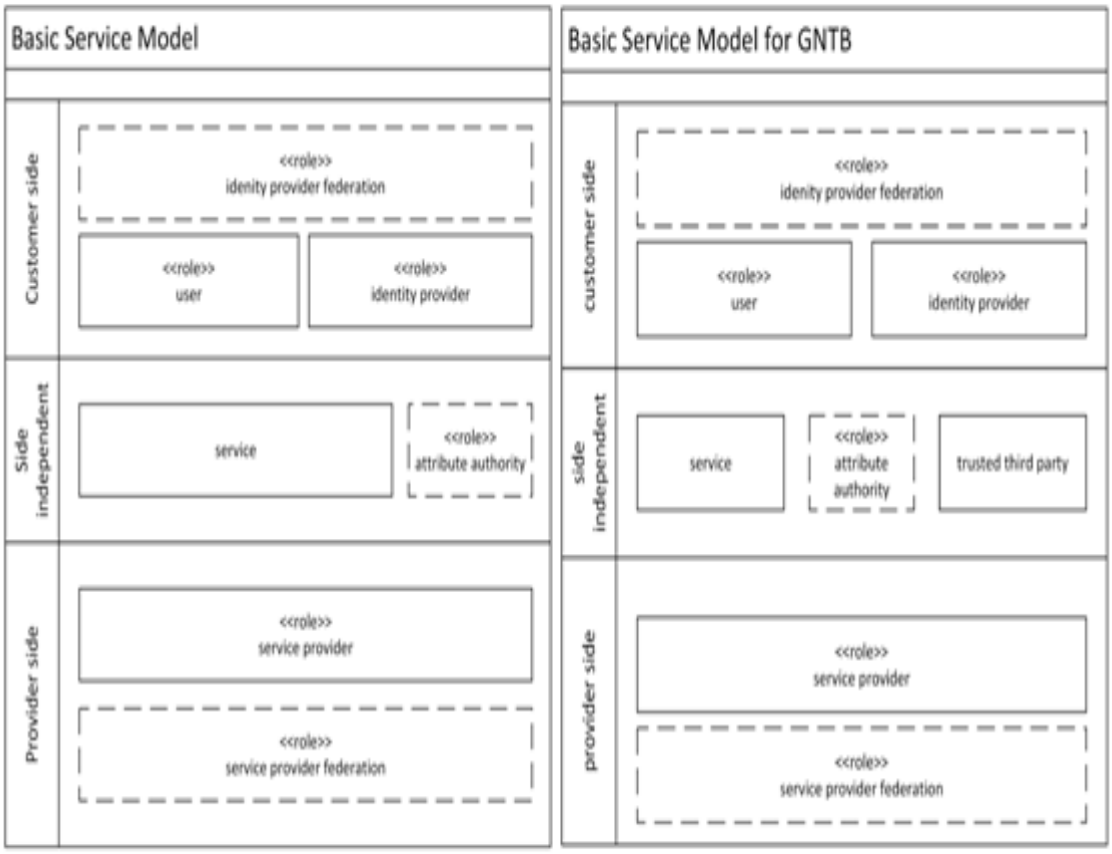


Figure 1: Basic Model for Federated Identity Management (a) without and (b) with a Trusted Third Party

The IDP side includes the user, the IDP, and the customer federation. The user accesses a service through the service client, which is normally a web browser. Intermediate, the service website of the original SP exists. The user information is stored in the local I&AM of the IDP and then passed to the IDP software, e.g., Shibboleth or SimpleSAMLphp. The provider side is basically not changed. An association between SP and service provider federation exists. Theoretically, an SP can have several service provider federations. The SP offers a service, which also makes use of the TTP.

The side independent part is modified in that respect that it makes use of the TTP. The functionality and the QoS parameters of the original service are not changed. As a result, we do not specify them in detail. The TTP is the service, which is used by the user of the localization service. The TTP is involved during the first time contact between the user and the service in order to exchange the metadata. Therefore, the TTP has a connection to the localization service, which is used to localize the user's IDP. The localization service supplies the external service access point of the original SP. The TTP is used by the IDP and SP to mediate service agreements. The TTP organization provides the implementation and the service of the TTP by operating the service. The TTP is therefore another service in the service view, which leads to the following:

- As SPs and IDPs actively use the TTP, they have access to the management functionality. The TTP management functionality provides information about the established connections, which can be used for further statistics and state reports. Since the customers need to be able to manage their registration and configure their level of trust at the TTP, this kind of management functionality is part of the service view. As the TTP is always involved during the first contact and different attack vectors apply, the TTP and the communication needs to be as secure as possible.

- The usage functionality is the initiation of the metadata exchange and the information for the user about the status of the exchange. From the customer's point of view, the management of the metadata exchange is a core functionality. Basic QoS parameters can be specified as availability, accessibility, and the metadata exchange time. The security properties are further parameters.

- Besides the service access point for the user, which redirects the user to the localization service, the customers use a web frontend respectively the extension of their SAML implementation, which is needed for the communication with the TTP. The web frontend functionality consists of the required functions for metadata management and account management. These functions can be used by the extension of the IDP/SP software. The extension furthermore automates predefined workflows. The security of these components is crucial as well.

- After specifying all details of the service, the service agreement needs to be presented.

The realization view in Figure 3 describes the realization of a service from the provider’s point of view. The service in this case is the TTP itself. The hierarchical relations between the services are developed in this view. The service could rely on, e.g., 2nd-level support. As no sub-services are implemented, no corresponding sub-service clients are needed. In addition to sub-services, a TTP provider operates and maintains the service. The information stored in the database and file-based data system is the main resource of the TTP. Another aspect is the service logic; workflows coordinate the usage of the resources. This leads to the realization view as described in the figure. The stored information is managed by the basic management functionality. Both SP and IDP make use of the functionality. The service logic, especially the metadata exchange, acts as user. The user wants to use a service, therefore the metadata needs to be exchanged. The user utilizes the service client to initiate the service logic, which is implemented.

The basic service model as well as both views visualize the TTP being another service within the FIM environment. The TTP interacts with IDP, SP and the user. If federations want to manage their federation via the TTP, they interact with the TTP as well. A TTP provider operates and administrates the TTP. As the TTP needs to provide management functionalities, e.g., in order to securely manage the metadata and trust information, the management architecture is shown in the next section.

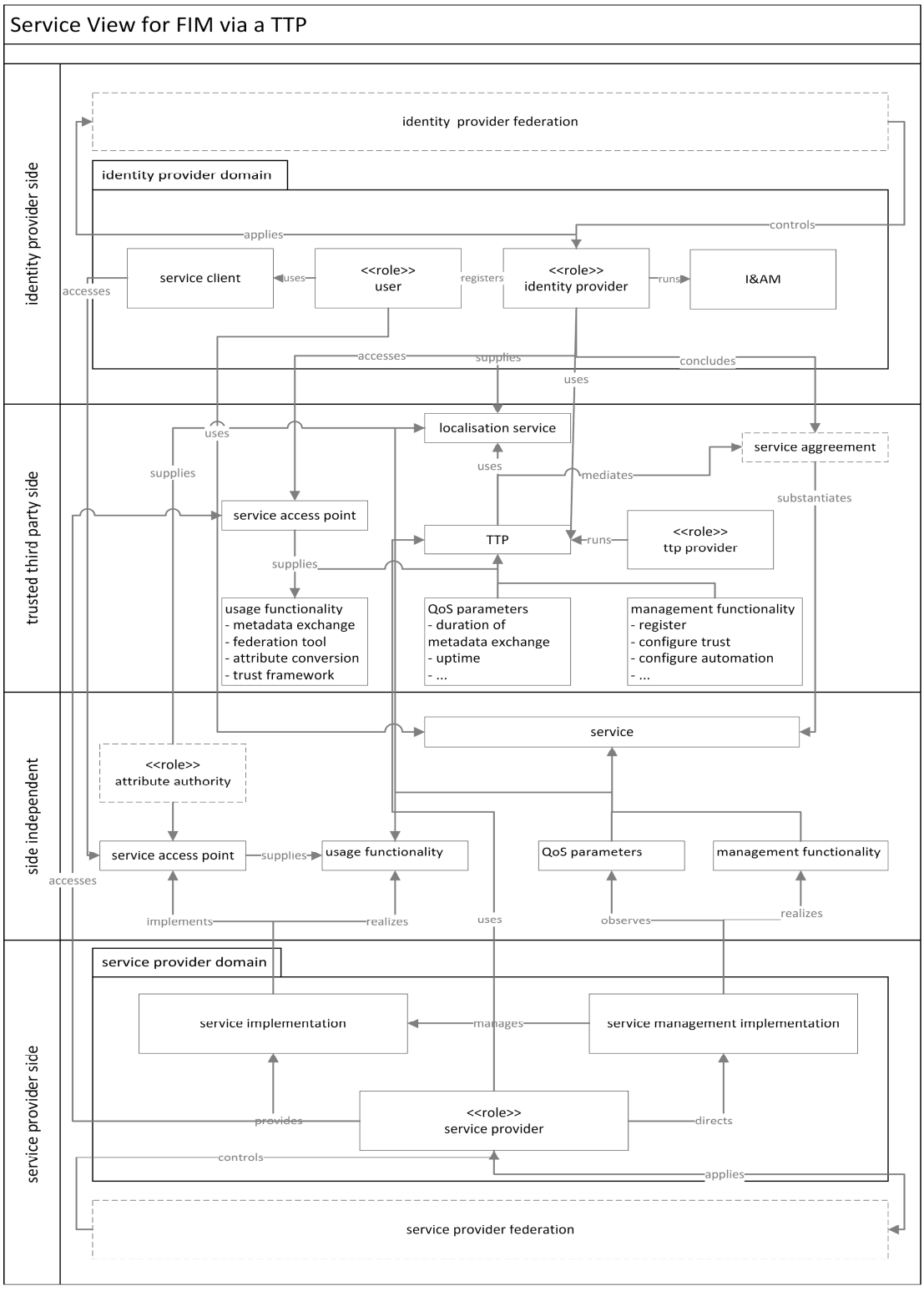


Figure 2: Service View for Federated Identity Management via a Trusted Third Party

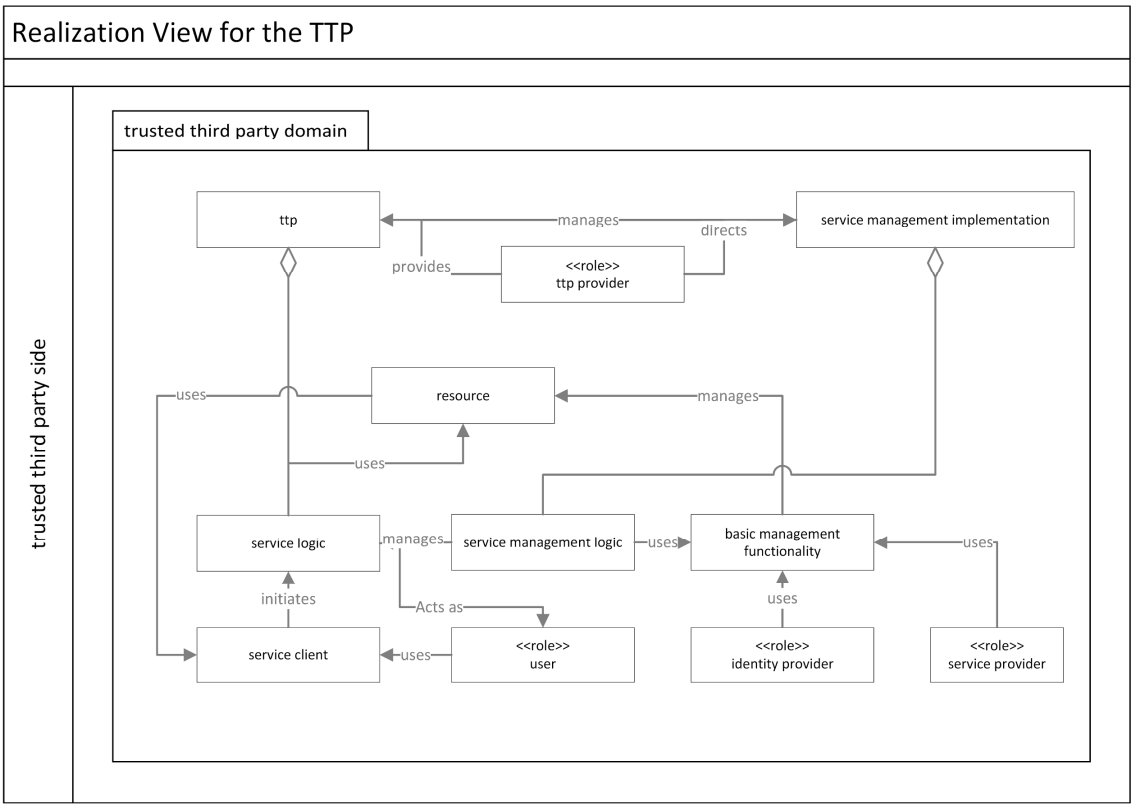


Figure 3: Realization View for the Service Trusted Third Party

4. FIM MANAGEMENT ARCHITECTURE

Standardized management architectures, which are derived from the dimensions of IT management, support the combination of management modules. These architectures are specified by different models [6], which need to be considered when implementing a TTP for dynamic automated metadata exchange:

- The information model defines a standardized format for management information.
- The organizational model specifies the roles of involved systems and their domains. Policies can be part of the organizational model describing targets or processes for technical management. This is shown above in the basic service model.
- Principles and concepts for the exchange of management information between the roles are defined in the organization model. Furthermore, the way of communication is defined, which is then realized by management protocols.
- The functional model divides the functions of the management in sub-functions, which are specified as basic management functionalities. These functions are normally implemented by management platforms and agent systems. The functionalities can be then used by the defined application programming interface (API).

As the security of dynamic metadata exchange is highly relevant for the operation of the service, it needs to be regarded in all models. These models are described for the management platform for FIM via a TTP in the following subsections.

4.1. Organizational Model

The organizational model describes which management domains are necessary for the management architecture with which roles and how these roles interact. Based on the service model for FIM, the domains SPDomain, IDPDomain, AADomain, fedDomain, and interfedDomain can be derived. The SPDomain is the management domain for the SP, representing the local domain of the SP, which provides the original service. The IDPDomain is the local domain of the IDP, which manages the user information. The AADomain is responsible for further user information, while fedDomain and interfedDomain represent the domain of a federation respectively inter-federation. These federation/inter-federation domains can have different structures, from an ad-hoc federation and hub-and-spoke federation to an identity network with different local coverage and differing trust models.

The domains contain several roles. The IDPDomain has *user* as a unique role. AADomain, SPDomain, and IDPDomain include a *general administrator*, which runs and configures the local software. A *relationship manager* is responsible for cooperation, e.g., with the federation and service providers. The *service desk* as the third technical role at AADomain, SPDomain, and IDPDomain is the contact point for incidents and problems. There are probably more roles within the organizations, which are not directly involved with the management platform and therefore not regarded.

The domains fedDomain and interfedDomain have, additionally to relationship manager, administrator, and service desk, several further roles, which are important for the platform. Federations and inter-federations are at some point initiated, either because of a project, a long-term cooperation, like in virtual organizations, or because of FIM as a general purpose. Therefore, the *initiator* is an additional role, which can initiate a federation at the management platform. Later, the role can pass the federation to a *general manager*, which is in control of it. If the service desk cannot solve a problem, technical *specialists* might be asked. The federation needs to be configured, e.g., the trust level needed to participate has to be set, by a *configuration manager*. As changes might have larger impact, a *change manager* is established as an additional role. These different roles have only the needed permissions to fulfil their job. The authentication should be done via SAML. For users without IDPs, local accounts have to be set up.

These specified roles interact via certain interaction channels, which need to be secure. While the administrators are in the background, all problems and incidents are communicated via the established service desks. This methodology is compliant with IT service management good practices such as ITIL and ISO/IEC 20000-1. Relationship managers should first interact via the service desk, though this is not likely to happen in reality. For federations participating in inter-federations and not with single other entities, the interaction is directed via the federation. The management platform can therefore serve as a united communication platform. This also means that the management platform needs the functionality and information for this task.

4.2 Information Model

The relevant information exchanged between domains within the management architecture is designed in the information model. It also specifies different domains and the format of information and resources. The goal of provisioning a management architecture is to handle federations respectively inter-federations as managed objects (MOs) and to automate the metadata exchange via a TTP. The specified information model defines MOs, which are relevant for the management, and their relationships. The definition has to take the expansion of the management platform into account, in order to be able to provide additional functionalities

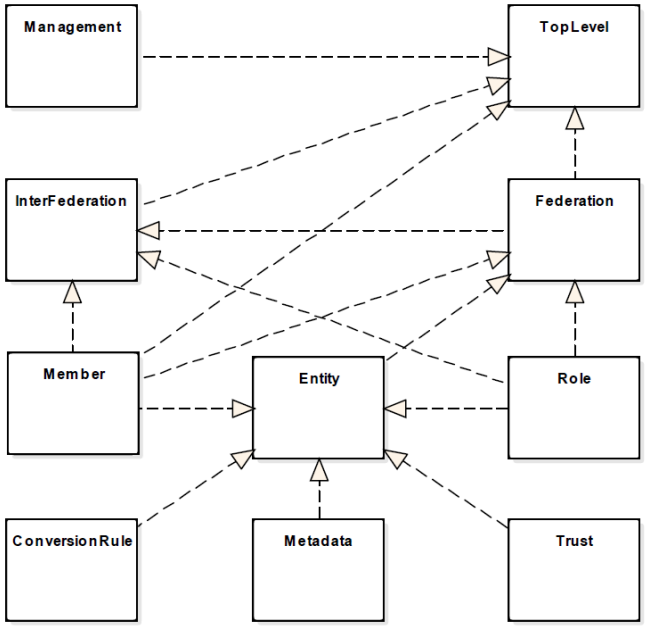


Figure 4: Domains of the Information Model

The information model for the FIM management architecture is shown in Figure 4. The domain specification contains the specifications at large, while the domain TopLevel represents the most generic class of the model. TopLevel contains different root elements and abstract root classes. These root classes, called domains, and dependent domains are the following: federation, inter-federation, entity, member, metadata, trust, conversion rule, and management.

As seen in the organizational model, federations, inter-federations, and entities contain different roles. The managed object federation can have different characteristics. Federations and inter-federation traverse the life cycle; a federation is first initiated, then it is operated, adapted, before it is closed. The representation of the federation is done via the domain federation, including all information about the structure and roles. Each federation consists of several entities, i.e., IDPs, SPs, and AAs. These entities are combined within the model domain entity. An organization can run several entities. Entities might be members of one or several federations. Entities need to exchange metadata, in order to establish technical trust. Entities need to register at the management platform as a prerequisite to exchange metadata. Given that federations consist of different entities, the membership of entities is shaped in the domain member. The domain is responsible for applying, verifying, accepting or denying, adding, and changing of memberships. Entities can exchange metadata independently of federations, if a user wants to make use of a

certain service. In order to exchange metadata via the management platform, the metadata needs to be managed. This is the purpose of the domain metadata. Whether SP and IDP are allowed to exchange metadata on demand depends on the risk and the trust in the partner entity. To calculate the trust in the other entity, different tools and standards can be used, such as the levels of assurance (LoA) paradigm. The trust in entities is modelled and compared in the domain trust. In order to understand the user information the IDP has, IDP and SP have to find a common syntax and semantics. In practice, IDPs have to convert the user information, also called attributes, into the SP's format. The domain conversion rule converts user information into the required format. Federations are, as described, seen as MOs. For the management of federations, policies and application processes are important among other things. These aspects of the management are described in the domain management.

4.3 Communication Model

The communication model specifies which entities communicate management information in which format and what communication mechanisms are used for management interventions, monitoring, and asynchronous notifications. The model also described additional services needed to support the communication mechanisms and the embedment of management protocols into the architecture and FIM.

Typical methods for communications are *post*, *get*, *set*, *query*, *create*, *delete*, and *update*. Register is needed to register a management object, while notify is used to inform other management instances. Furthermore, *discover* identifies management instances. These methods are then transferred into workflows, protocols, and an API to provide the needed functionality (described in the next section) and to communicate and interact, as described above. The protocols should be transformed into SAML, as it is the predominant standard for R&E and commercial sectors. The methods need to be able to securely exchange metadata, trust information, and conversion rules. Authorization and publication/discovery of interfaces for interaction channels is another important functionality. The interaction channels need to support loose as well as defined structures of cooperation, while different actions and activities are logged.

The method register is explained as an example. In order to communicate over an interaction channel, the different roles and entities need to be aware of each other. As prerequisite, all entities need to register at the management platform. The registration is the basis to publish a service and therefore make the own service, SP service or IDP user information, available for other entities. Registration is also important for federations and inter-federations, as they can use the management platform as management tool. In this example, an entity registers at the management platform, here described as TTP. It is then verified, e.g., by a certificate. Afterwards, the entity applies at a federation, which first verifies the entity. This might be done automatically, dependent on the application process. If the federation approves, the entity becomes a member of the federation.

4.4 Functional Model

The functional model structures the management into different functional areas and establishes common management functionalities. The goal is to determine generic functional components, which are required for the management of federations and FIM.

The entity level contains IDP, SP, and AA, while the federation and inter-federation level comprises the same functional areas. The functional area conversion rule management is relevant for both, IDPs and AAs. Conversion rules can be created, changed, deleted, downloaded, and validated. As the entity might not want to download and integrate conversion rules automatically, the degree of automation can be configured. Metadata management contains the upload, change, deletion, exchange of metadata, and the configuration of the degree of automation. Also notifications and logging are important. Users as well as entities should be notified, if the metadata exchange fails and the user, when he successfully can use a service. The configuration management handles the configuration with generating, changing, and deleting a configuration. The configuration refers, in this context, to the configuration of both, (inter-)federation and entities.

Besides the technical trust, established by exchanging metadata, further aspects of trust, like behavioural trust, might apply. The trust value can be set, verified, configured, and compared. For example, a SP requires a certain trust level for its service, which depends on the risk for the service. Therefore, the administrator configures the required trust level. When a user of an IDP wants to use a service, the trust level of the IDP is compared to the configured trust level, before the metadata exchange takes place. If the trust level of the IDP is high enough, the metadata exchange takes place and the user can immediately use the service. Otherwise, the administrator of the IDP and the user get a notification about the problem. The most primitive case of trust is some sort of black and white list, while different LoA schemes might be used as well.

By applying for membership in a federation, an entity accepts the use of the federation's policy. Policy management is one functional area at the federation/inter-federation level. The policy, written in XML, should allow a mostly automatic validation of the entity. A parser should counter-check the elements in the entity's metadata, if possible. The relationship manager of the (inter-federation) first has to create such a policy. Later onwards, he can change and delete it. When the policy is changed, members of the federation need to be notified and checked again against the policy. If conflicts arise, the relationship manager needs to solve them. Policy management is closely related to member management, where federations manage their members. This functional area contains the query of member, memberships, and role information. Memberships might need to be changed, applicants accepted or denied. Roles for the administration tool can be changed, and notifications for changes should to be sent. The policy as well as the application process have to be determined, while important actions are logged.

Another functional area for the federation/inter-federation level is service management. The general managers of federations and inter-federations need an overview of all provided services and their usage. While this information has to be queried and visualized, the management information also can be sent to members, if appropriate.

5. FIM SERVICE MANAGEMENT

In order to discuss how service management, the management architecture, and the management platform can improve the TTP, FIM service management is explained in detail. Many IDPs and SPs run their own scripts to parse SAML software log files for relevant events in order to gather data for statistics of service usage. Additionally, a few tools for monitoring SAML entities exist. To create a comprehensive overview for a federation and its members, a unified aggregation

architecture needs to be developed, which suits most implementations. In order to ensure this, this extension is separated from the provider implementation.

An overview of all services and their usage as well as generic information about federations and inter-federations should be displayed. Further interesting facts for administrators are:

- Number of participants, IDPs, and SPs
- Number of metadata exchanges
- Number and ranking of used services by popularity
- Overview of trust within a federation
- Overview of conversion rules and their usage
- Overview of technical information
- Overview of queried user attributes in general and per service entity category

The query about service management should be done by a defined method and displayed via a dashboard, to which only administrators should have access. The dashboard should show the numbers but also visualise them and their relationships. For example, the established trust relationships between IDPs and SPs can be shown on a map or globe, allowing information about the dimension of the federation and the trust to be viewed in different scales. While the overview might give an idea about the dimension, a more detailed view shows the entities and the problems of trust, while a drilled-down view shows detailed information about the entity. This information can be used, for example, to facilitate inter-organizational security incident management when users misused their permissions or SPs have gone rogue.

Additionally, the trust information can be encoded by colours, in order to display errors during the metadata exchange, but also other categories, such as:

- IDPs vs. SPs
- Different federations or other pre-established trust boundaries
- Levels of Trust applied by the SAML entities
- Schemas that are used for user attributes
- Age of metadata and X.509v3 server certificates as well as timespan until the certificate expires
- Period of membership
- Number of successful metadata exchanges

The expiration of the certificate can be a source for errors, if metadata with expired certificates is not processed by an IDP or SP.

In order to visualise this information, it needs to be collected beforehand. The collecting server can be the TTP. Additionally, the following components are required:

- Provider: IDP or SP, which provides some kind of statistics. Usually, the data can be read from or determined by parsing SAML software log files.
- Agent: The agent software is used to read, parse, filter, and relay the log events generated by the provider. The agent supports multiple destinations and can apply different filter rules.

- Aggregator: A central instance that aggregates the statistics sent by the providers via the agents.
- Web frontend: Method of displaying the generated statistics.

The agent contains parsers for the specific SAML software in order to determine metrics by parsing log files, reading status pages, or other methods. Filtering is based on the configuration of the responsible administrators. Furthermore, the log files have to be written in a standardised format. This is done by a defined syntax. By applying different filters and aggregators, the administrators can differentiate between detailed local and general federation-wide statistics. After authenticating, the agent's filtered performance data is stored in a database, based on the ID of the provider, the type of the event, and a timestamp.

In order to display statistics, the web frontend requests data from the aggregator's database. The returned data is filtered according to the scope of the user's request. Depending on the configuration of the client, the event is then sent to the aggregator immediately or interval-based, which in turn informs the web frontend of the updated data and refreshes the display in the user's browser. The statistics can be a functionality of a portal, combining all the different functionalities from the functional model.

6. CONCLUSION AND OUTLOOK

Dynamic automated metadata exchange (DAME) for FIM enables the on-demand, user-triggered exchange of SAML metadata between IDPs and SP across current federations' borders. The scalability of the metadata exchange in federations and inter-federations is improved at the same time, as only the really necessary metadata is exchanged. It therefore increases the automation and scalability of formerly manual implementation steps by administrators. Consequently, the users can immediately use a new service without extended waiting periods due to the involvement of administrators.

In order to create an overview of the changed architecture and the service of the TTP, the MNM service model was applied to both, traditional FIM and FIM via a TTP. The MNM service model helps to distinguish between customer and service provider, and gives a neutral view on the service. The service view for FIM via a TTP added another side, the trusted third party side, to the view, as the service TTP is an additional service. The TTP can be provided by a federation or inter-federation operator, helping members to connect to the outside world.

Based on the service view and the realization view for FIM via a TTP, the management architecture was designed. While the technical TTP helps to exchange metadata on demand, the management architecture describes functional areas relevant for FIM. Federations and entities can use the management platform to manage metadata, members and make use of further functional areas. The approach of dynamic metadata exchange with the addition of a management platform, as described in this paper, allows for example the configuration of trust, reducing the risk of data loss and prohibited usage of services. Federations and inter-federations can make use of the management platform to manage their members, while entities have a tool to manage metadata and conversion rules. The organizational, information, and communication model describes the information, interactions, interfaces, and roles within organizations, which need to be regarded during the next step, the design and implementation of the management platform. This leads to an extended TTP, helping organizations to manage FIM and gather statistics about their FIM usage.

Further research topics relate to a detailed security and risk analysis of such an extended TTP as well as the trust between two entities. Though the technical trust is exchanged via the metadata, the quality of the entity could be assured or estimated by a level of assurance and dynamic trust. Furthermore, visualization of information should be regarded in more detail, especially with the focus on inter-organizational security incident management.

ACKNOWLEDGEMENTS

The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful comments on previous versions of this paper. The MNM-Team, directed by Prof. Dr. Dieter Kranzlmüller and Prof. Dr. Heinz-Gerd Hegering, is a group of researchers at Ludwig-Maximilians-Universität München, Technische Universität München, the University of the Federal Armed Forces, and the Leibniz Supercomputing Centre of the Bavarian Academy of Sciences and Humanities.

REFERENCES

[1] Cantor, S., Kemp, J., Philpott, R., and Maler, E.: Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Security Services Technical Committee Standard (2005)

[2] GÉANT: eduGAIN technical site. https://technical.edugain.org/status.php [Online; 12.01.2016]

[3] Hommel, W., Metzger, S. and Pöhn, D.: Géant-TrustBroker: Dynamic, Scalable Management of SAML-Based Inter-federation Authentication and Authorization Infrastructures. ICT Systems Security and Privacy Protection, Springer Berlin Heidelberg (2014)

[4] Young, I., Ed.: Metadata Query Protocol - draft-young-md-query. http://datatracker.ietf.org/doc/draft-young-md-query/ [Online; 12.01.2016]

[5] Pöhn, D.: Dynamic Automated Metadata Exchange - draft-poehn-dame. https://datatracker.ietf.org/doc/draft-poehn-dame/ [Online, 12.01.2016]

[6] Hegering, H.-G., Abeck, S. and Neumair, B.: Integrated Management of Networked Systems - Concepts, Architectures, and Their Operational Application. Morgan Kaufmann Publishers (1999)

[7] Garschhammer, M., Hauck, R. and Hegering, H.-G. et al.: Towards generic Service Management Concepts - A Service Model Based Approach. Proceedings of the 7th International IFIP/IEEE Symposium on Integrated Management (IM 2001)

[8] Garschhammer, M., Hauck, R. and Kempter B. et al.: The MNM Service Model - Refined Views on Generic Service Management. IEEE Journal of Communications and Networks, vol. 3, no. 4, pp 297-306 (2001)

[9] Garschhammer, M., Hauck, R. and Hegering, H.-G. et al.: A Case-Driven Methodology for Applying the MNM Service Model. Proceedings of the 8th International IFIP/IEEE Network Operations and Management Symposium (NOMS 2002)

AUTHORS

Daniela PÖHN received a university master degree in Computer Science from the University of Hagen, Germany, in 2012. She was engaged in the IT industry as a full-time software developer during her studies, before she joined LRZ as a Ph.D. candidate in September 2012. Her main research focus is on identity management.

Wolfgang HOMMEL has a Ph.D. as he teaches information security lectures and labs. His research focuses on information security and IT service management in complex large-scale and inter-organizational scenarios.

SURVEILLANCE VIDEO BASED ROBUST DETECTION AND NOTIFICATION OF REAL TIME SUSPICIOUS ACTIVITIES IN INDOOR SCENARIOS

Nithya Shree R, Rajeshwari Sah and Shreyank N Gowda

Department of Computer Engineering,
R. V. College of Engineering, Bangalore, India
nithyashree675@gmail.com
rajeshwari.sah@gmsil.com
kini5gowda@gmail.com

ABSTRACT

Over recent years, surveillance camera is attracting attention due to its wide range of applications in suspicious activity detection. Current surveillance system focuses on analysing past incidents. This paper proposes an intelligent system for real-time monitoring with added functionality of anticipating the outcome through various Image processing techniques. As this is a sensitive matter, human decisions are given priority, still facilitating limited logical intervention of human resource. This framework detects risk in the area under surveillance. One such dangerous circumstance is implemented, like a person with a knife. Here the prediction is that in the firm places like ATM, Banks, Offices etc. a person possessing knife is unusual and likely to cause harmful activities like threatening, injuring and stabbing. The experiment demonstrates the effectiveness of the technique on training dataset collected from distinct environments. An interface is developed to notify concerned authority that boosts reliability and overall accuracy.

KEYWORDS

Surveillance behaviour; Real-time dynamic video; pattern match; precautionary measures; sharp objects;

1. INTRODUCTION

The rapid growth of criminal cases have increased the need to establish image processing technology in security based system. Surveillance camera is an integral part of any threat monitoring system in various public scenarios like conferences, shopping mall, restaurants, community gathering etc. Hence, surveillance input plays an intuitive role in abnormal activity detection. Moreover, digital equipment such as Web camera, processing machine instance and hard disk drive are mass-produced, and are sold at low price [2]. For analysis of real time events, multiple frameworks are proposed; they include tracking, learning and monitoring surveillance footage [1-4]. Such analysis concentrates on factors like human posture [5-8] [13], hand

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 227–236, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60618

movements [9-12] and object properties present in the video frame sequences [8]. To estimate the possibility of danger caused by any physical force is well identified in this paper by recognizing the sharp and harmful weapons, hand prehensile and grasping movements. A person being attacked by a knife in an environment of more than one individual is the major concern of our research project.

The proposed framework is a novel approach to detect and notify indoor violence like stabbing, thrashing or any activity involving physical force, to the concerned authority. The implementation is useful for examining such suspicious activity in enclosed places like ATM, Classroom, theatre, houses etc. A drastic change is brought in the society by introducing technology in enhancing the security system prevailing in the present world. The system is also dynamic in sending message to the relevant authority for undertaking precautionary measures or enforcing the desired action.

2. RELATED WORK

The influencing factor to the experiment is the benefit of social security that is foreseen by enhancing the functionality of Visual surveillance camera by embedding the provision of processing, detecting and notifying the menace befallen by the person who is captured in its input video.

We are interested in the previous work that come from the field of human interaction and motion tracking in order to qualitatively analyse the real time human motion and predict the ongoing activities in contact with other human being present and with the objects. Most of these work use background subtraction practice to get the binary foreground image. [5][6][3].

A vast amount of literature exists on real time hand tracking. Few are picked as follows: Javeria Farooq et al. [12], Milad Rafiee Vahid et al. [16], P Raghu Veera Chowdary et al. [9], Mykyta Kovalenko et al. [10] and Pedro Cisneros et al. [11] provide information to the logic of the experiment for capturing the real time hand movement data and processing them using matching, computing and anticipation techniques to get the actual hand behavioural understanding that helps in decision making of the scenario under test. We are interested in works from pattern recognition [13] and comparison for identifying the hand posture and comparing it with the data base images [14]. Also, the knowledge with respect to hand prehensile and clutching movements are acquired that is used in the apprehensive area for identifying the hand mould to conclude the presence of object in hand.

Although, Human activity processing and hand-object interaction analysis is a difficult task from a machine vision perspective it has extensive benefit to the Societal security and brings a drastic awareness in the current system to prevent offensive conduct in public space.

The approach is to recognise the presence of sharp object in hand which is one of the means to cause harm. The shape, orientation and projection of the hand held object is to the discovered. Radu-Daniel Vatavu et al. [15] conducted an investigation on the feasibility of using the posture of the hand during prehension in order to identify geometric properties of grasped objects such as size and shape. Garg et al. [8] suggested an image segmentation technique to recognize the hand object in a scene using Mathematical analysis and locate the object position in the Scene. However most of the work have concentrated on hand holding objects in general but not

specifically sharp and harmful objects like knife. We intend to design a model which identifies such cases and also notify in case of any threat.

3. METHODOLOGY

The block diagram of proposed suspicious activity detection framework is shown in Fig 1. The proposed framework consists of two main components: 1. (FBD) Framing and Blob detection (Input video processing) 2. (HON) Human tracking, Object (like knife) identification and notification (query Image processing) stage.

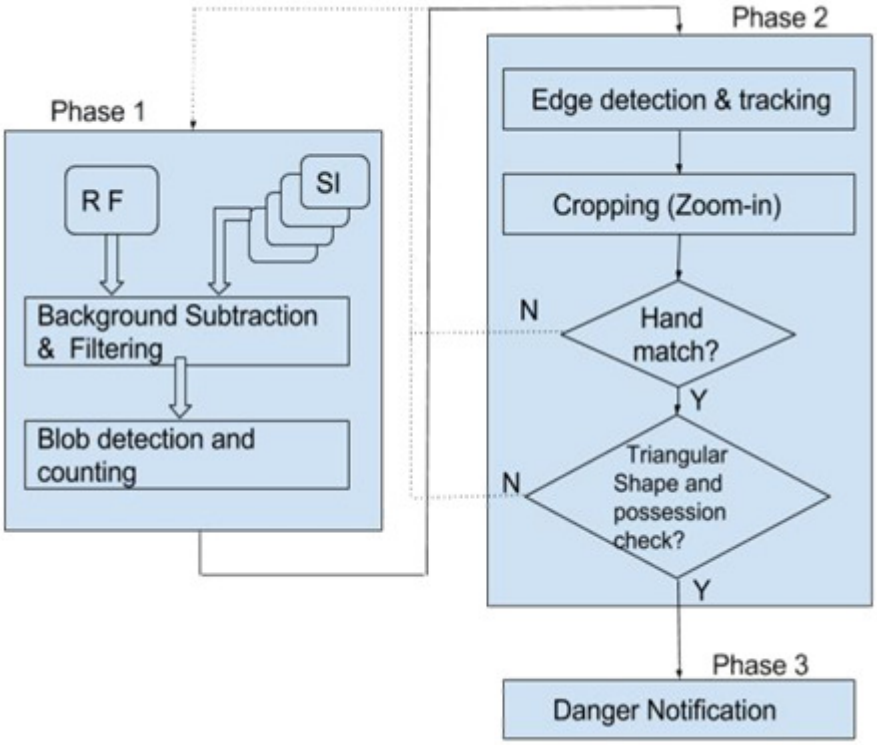


Figure 1. Block Diagram of complete algorithm

In the Framing and Blob detection stage as shown in Phase 1 of block diagram in Fig. x, Frames are extracted from the input surveillance video and compared with respect to the reference background image. Also the blob detection and analysis is done to calculate the number of blobs (i.e., human being in concern). If the number of blobs are greater than one then the program enters the next section i.e., Phase 2 of the block diagram shown in fig. x, otherwise the execution flow comes to an end for the particular frame and continued with next consecutive frame in the video.

In the Human tracking, Object (knife) identification and notification stage, edge detection helps in observing the characteristics of human blob present in the scene. The frame is zoomed-in concentrating the blobs by cropping the frame into certain dimension and is further divided into 'x' regions (and stored in the current working directory). Every block is further processed for accuracy.

Object detection returns positive result if the three conditions are satisfied; 1.Hand mould and gesture verification 2.Presence of Sharp object affirmation and 3. Link between the hand and object to meet the desired soleness. If the scenario is validated to be insecure, the program flow reaches the Phase 3 of the block diagram shown in fig. x in which a notification along with relevant photograph or latest timestamped part of video is sent to the concerned security system like nearby Police department, guard, NGO or anyone who can take quick action.

3.1 Framing and Blob detection (FBD)

The input video is taken from the surveillance camera with the Camera specification being 2MP. The video is sliced to form frames and is stored in a directory 'frames' within present working directory. A precise initial frame n1 is taken as a base frame with reference to which the further analysis of the video is done, it is basically the background image.

The looping construct is used in order to traverse the complete set of frames of the video with Σi frames where i from 2 to number of frames (n) and each frame is inspected against the base frame. Each frame undergoes the blob detection step. Blob detection is achieved through typical background subtraction between the current frame and background base frame. Morphological operations are performed in order to remove the holes, noise and unneeded information from the subtracted image indicates the number of individuals in the frames which can be determined by the area calculation method where in a defined threshold is set for the white fields appearing on the image to become blobs. If a particular white patch meets the threshold set then that particular object is considered as blob. If the number of blobs in a frame exceeds 2, then the further processing is continued. Otherwise the process halts anticipating that no hazard can take place as there is only one person, hence returns to continue with the next frame in the sequence of the video consecutively. The algorithm is efficiently implemented in order to optimize the resource usage.

3.2 Human tracking, Object (knife) identification and notification (HON)

Edge detection using Sobel's algorithm helps in human being tracking and systematic user interface design. The blobs are cropped to ignore the unnecessary processing of complete frame. The cropping is done using a set of if-else-if ladder along with the decision making factors being brute force length-width ratio calculation having known the centroid of the blob, originate pixel and area. The cropped blob is stored in the current working directory of the running program.

The blob concentrated image is further divided into multiple regions to achieve high accuracy and is accessed from the same PWD (present working directory). The cropped images are further taken for analysis.

The next step takes the image parts for processing and checks to see if these 3 conditions are contented. Only if the 3 below noted criteria's are met, the system tends to notify the desired end user and further action is left to the security system in charge. The conditions are as follows:

1. Hand mould and gesture verification: The hand posture is evaluated by matching with the data base images in the repository in complete 360 degrees to see if it is of clenched fist form. This phase concludes that the hand is holding some object.

2. Triangular or sharp object detection: The block of cropped image is checked to identify sharp object present in the frame. This is achieved by performing triangular object detection test on the frame by applying certain range of threshold condition on the length-width ratio, type of triangle, orientation of triangle etc. This ensures the presence of sharp object (knife like object) present in the frame.

3. There should be a link between hand suggested by condition 1 and possessed object suggested in condition 2. This framework is not concerned about the presence of sharp object alone, it should be held by any of the person in the frame. This can be verified by the logic with soleness check of the object in the image, i.e., there should be a single speck in the binary frame to confirm that the object x is possessed by the person only.

If all the conditions are satisfied and the flow of execution reaches till the end overcoming every decision making construct, then it can be concluded that there is presence of danger in the scenario and hence the function for notification is invoked by the main program. As the considered security topic is sensitive, the framework defined also needs human intervention for exact decisions. But human resource is well utilized by having less interposing work. Rather is required to handle only when the system sends the notification.

Similarly all the frames in the captured video are analysed to detect suspicious activity and evaluated for the secureness of the real time scenario.

3.3 Workflow

1. Capturing input video and slicing into frames by various Matlab inbuilt functions such as NumberOfFrames, resd(VideoObj, frameno), imwrite() etc. Identifying reference background frame.

2. Traversing each frame of the video and analyzing with respect to the considered reference frame. If there is danger identified, then it is notified and paused for the response.

3. Adaptive background subtraction and noise removal by Image filtering methods available in Matlab and other predefined filters.

4. Blob detection and counting by Connected Area Component labeling technique. Break and start processing next frame if less than two people or blobs. Continue if more than 2 blobs detected.

5. Edge detection by means of Sobel edge detection technique for enhanced user interaction and human tracking.

6. Notifying through the relevant means if all of the conditions are satisfied: hand mould is matched, triangle like sharp object present in the frame and link existing between hand and object.

4. EXPERIMENTAL RESULT AND ANALYSIS

The proposed suspicious activity detection system is evaluated on real time recorded data set for validation of framework's procedural tasks. The video database includes 10 hours of video covering a wide variety of content. The format of reference video clips is 1280*1024 pixels and 30 frames/ sec. In our experiments, video clips that are taken as input are selected from reference dataset that are captured in 5-6 distinct locations.

The experiment conducted in three different environment as shown in Fig. 2 is taken for analysis. The first scenario is a well-lit classroom with benches, fans, curtains etc. as shown in fig. 2(a). The second scenario as shown in Fig 2(b) witnesses the experiment conducted in a university computer laboratory. The third case is a moderately-lit shuttered indoor space as shown in fig. 2 (c). Video clips shot in all the three locations is 30 to 45 seconds long per instance.

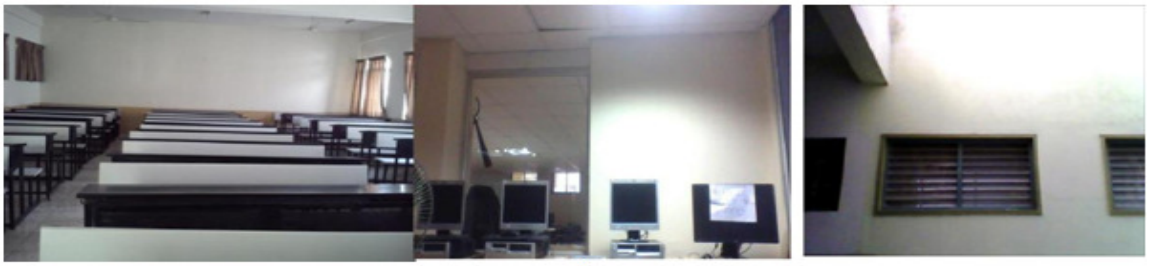


Figure 2. Experimental Input locations

The possible threat situation for the above three cases is shown in Fig. 3 First image depicts a scenario in which a person is trying to attack another person reading a book in a class, with a knife behind her back Fig 3(a). In the second image Fig 3(b) a person is pointing a knife towards the other which accounts for a threat situation. Fig 3(c) shows a person holding a knife in a shuttered indoor space, very close to another individual which might be a suspicious situation.

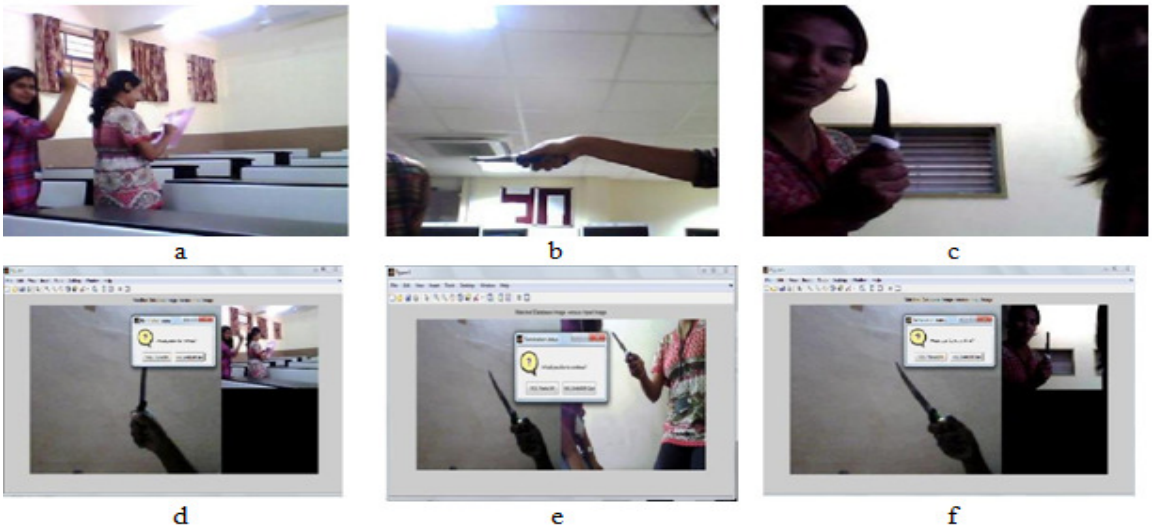


Figure 3. Threat Scenarios and their respective notifications

The possible threat in all the above three cases are notified by a pop up window shown in fig 3(d, e, f).

The entire process is done in sequence of steps. For the input image Fig 4(a) the different stages of the experiment is traced by the subsequent images in Fig 4. Fig 4 (b) shows the blob detection phase of the experiment where the two individuals are identified as two separate blobs. 4(b) is obtained by subtracting the Fig 4(a) with the reference background image with appropriate filters. The edge detection stage is shown by Fig 4(c) which is done using Sobel edge detection method with a defined threshold. The fourth image Fig 4(d) constantly tracks the foreground image in the video sequence. The tracking also helps in efficient implementation of user interface. For the danger notification which is the final step of the experiment, a pop-up window is displayed as shown in fig 4(e).

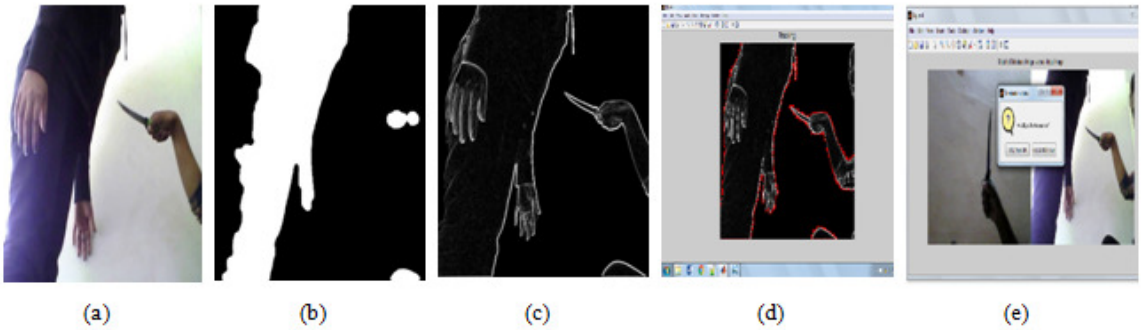


Figure 4. Different Stages of the experiment

Apart from all the true positives, the system also shows two false positives and three false negatives. For the shown fig 5(a) where the person is pointing a finger in the shape of a knife to a person in the vicinity gives a danger notification irrespective of the absence of the knife. The Hand shape is mistakenly seen as a knife by the system. In the second image shown in Fig 5(b) where the person's fist is falsely matched with the database image giving a false positive notification.

Figure 5. False Positives

The false negative cases is shown in Fig 6. In the first image Fig 6(a), no message is notified despite the presence of knife in one's hand. This is because the knife is being held in a different fashion for which system is not trained. Fig 6(b) also fails to notify any threat as the light intensity has caused shining glare on the knife which makes it undetected. In the third image Fig 2.5(c), the two blobs are overlapped along with the blob for the knife which makes it difficult for the system to detect.

a. b. c.

Figure 6. False Negatives

The system is evaluated by a precision and recall metric for the collected dataset from 3 different locations as mentioned in Fig 2.1. The particular rows represent the cases considered in Fig 2.2. It includes True positives, True negatives, false positives, false negatives count for the input video with 100 frames. Also Precision and Recall values are calculated. It is observed that the precision values decreases and recall increases with the subsequent test cases.

Table 1. PR Table of analysis result

Query Video	True Positives	True Negatives	False Negatives	True Negative	Precision (tp/(tp+fp))	Recall tp/(tp+fn)
1	61	0	30	9	1	0.67032967
2	41	10	12	37	0.803921569	0.773584906
3	65	25	7	3	0.722222222	0.902777778

5. CONCLUSION AND FUTURE WORK

This paper proposes a possible technique for Suspicious Activity Detection in indoor places to enhance the security system by improving the functionality of Surveillance camera to an intelligent device that is capable of detecting and notifying danger. An observer or the guard is relieved from the burden of continuous monitoring, may be physically or virtually watching enormous amount of video sequences captured by multiple Web cameras. Instead, intervention is serviced when the notification is sent. Due to its cost effectiveness, simple installation, scalability to different video resolutions, and once in a lifetime initialization, this is the feasible and practical solution to deploy in real scenarios.

The algorithm works efficiently in bright areas with 73 % accuracy, whereas functions moderately in less intensified areas with 67 % accuracy when experimented against the real time videos captured from distinct places.

Our goal is to rigorously continue to improve our detection and notifying system by adding various other features, such as implementation to detect other harmful activities like striking down, hitting, snatching the belongings and other physical abuse activities by supporting additional hand and object poses, expanding the dataset, experimenting in distinct possible locations, introducing machine learning techniques by using priori incidents knowledge with the focused concern of maintaining societal security. In the future, we expect promising results on

location specific distributional implementation with flattering performance. There is certainly room to improve the accuracy of the system to reduce the false positives and true negatives and to investigate on the category of activity and level of impact.

We believe, that the most rewarding impact is foreseen in security security however, the accuracy of the system needs to be enhanced for increased trustworthiness.

ACKNOWLEDGEMENT

We would like to thank Dr. Roopalakshmi, RVCE, and Bangalore for valuable discussions and many students of the University for creating vast test videos.

In addition, we would like to thank the anonymous reviewers for their valuable comments and suggestions that have helped to significantly improve the quality of this manuscript.

REFERENCE

[1] Hidetomo Sakaino, (2013) "Video Based Tracking, Learning and Recognition Method for Multiple Moving Objects", IEEE transactions on circuits and systems for video technology, Vol. 23, NO. 10, pp 1661-1674.

[2] Miwa Takai, (2010) "Detection of Suspicious Activity and Estimate of Risk from Human Behavior shot by Surveillance Camera", Second World Congress on Nature and Biologically Inspired Computing, pp 298-304, IEEE publishers.

[3] Tao Luo, Ronald H. Y. Chung & K. P. Chow, (2014) "A Novel Object Segmentation Method for Silhouette Tracker in Video Surveillance Application", IEEE International Conference on Computational Science and Computational Intelligence, Vol. 1, pp 103-107.

[4] Huang Li, Yihao zhang, Ming Yang, Yangyang Men & Hongyang Chao, (2014) "A rapid abnormal event detection method for surveillance video based on a novel feature in compressed domain of HEVC", IEEE International Conference on Multimedia and Expo(ICME),pp 1-6.

[5] Lulu Chen , Hong Wei & James Ferryman, (2013) "A survey of human motion analysis using depth imagery", Pattern Recognition Letters, Vol 34,Issue 15, pp 1995-2006, Elsevier publishers.

[6] Chengzhang Qu, Yuewei Lin, Dengyi Zhang & Song Wang, (2013) "Distant Human Interaction Detection from Kinect Videos", IEEE International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), pp 1372 – 1375.

[7] Manoranjan Paul, Shah M E Haque & Subrata Chakraborty, (2013) "Human detection in surveillance videos and its applications - a review", EURASIP Journal on Advances in Signal Processing, Springer Publication.

[8] Garg R.,Aulakh I.K & Kumari N. , (2014), "A mathematical model to detect hand object from the scene", IEEE International Conference on Advance Computing (IACC), pp 1133-1136.

[9] P Raghu Veera Chowdary, M Nagendra Babu, Thadigotla Venkata Subbareddy, Bommepalli Madhava Reddy & V Elamaran, (2014), "Image Processing Algorithms for Gesture Recognition using MATLAB" , IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), pp 1511-1514.

[10] Mykyta Kovalenko, Svetlana Antoshchuk & Juergen Sieck , (2014), "Real-Time Hand Tracking and Gesture Recognition Using Semantic-Probabilistic Network", 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, pp 269-274, IEEE Conf. Publications.

[11] Pedro Cisneros & Paul Rodr´ıguez, (2014) , "Practical Hand Tracking Solution by Alternating the Use of a priori Information", 2014 IEEE 5th Latin American Symposium on Circuits and Systems (LASCAS), pp 1-4, IEEE Conf. Publications.

[12] Javeria Farooq & Muhaddisa Barat Ali, (2014) "Real Time Hand Gesture Recognition for Computer Interaction", 2014 International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE), pp 73-77 ,IEEE Conf. Publications.

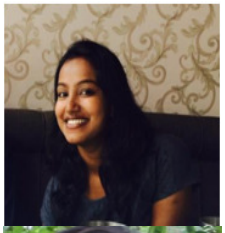
[13] David Minnen & Zahoor Zafrulla , (2011), "Towards Robust Cross-User Hand Tracking and Shape Recognition", 2011 IEEE International Conference on Computer Vision Workshops, pp 1235-1241.

[14] Van-Toi Nguyen , Thi-Lan Le, Thanh-Hai Tran, R´emy Mullot & Vincent Courboulay , (2014) , "Hand posture recognition using Kernel Descriptor", 6th International conference on Intelligent Human Computer Interaction, IHCI 2014,Procedia Computer Science, Vol. 39,pp 154-157 ,Elsevier Publications.

[15] Radu-Daniel Vatavu, Ionut & Alexandru Zait, (2013) , "Automatic recognition of object size and shape via user-dependent measurements of the grasping hand", International Journal of Human-Computer Studies ,Vol. 71,Issue 5,pp 590-607, Elsevier Publications.

AUTHORS

Nithya Shree R.



Students of R.V College of Engineering, Bangalore, India. I am interested in introducing technological enhanced solution for real time problems.

Rajeshwari Sah

Student of R.V College of Engineering, Bangalore, India. Most of my work are in the field of computer vision, image processing and machine learning domain

Shreyank N Gowda

Student of R.V College of Engineering, Bangalore, India. I am interested in the field of image processing, virtualization and cloud computing domains.

MAJORITY VOTING APPROACH FOR THE IDENTIFICATION OF DIFFERENTIALLY EXPRESSED GENES TO UNDERSTAND GENDER-RELATED SKELETAL MUSCLE AGING

Abdouladeem Dreder[1], Muhammad Atif Tahir[1], Huseyin Seker[1] and Muhammad Naveed Anwar[2]

Bio-Health Informatics Research Group
[1]Department of Computer Science and Digital Technologies
[2]Mathematics and Information Sciences
Faculty of Engineering and Environment
The University of Northumbria at Newcastle upon Tyne, NE1 8ST,
United Kingdom.
{a.dreder,muhammad.tahir,huseyin.seker,naveed.anwar}@northumbria.ac.uk

ABSTRACT

Understanding gene function (GF) is still a significant challenge in system biology. Previously, several machine learning and computational techniques have been used to understand GF. However, these previous attempts have not produced a comprehensive interpretation of the relationship between genes and differences in both age and gender. Although there are several thousand of genes, very few differentially expressed genes play an active role in understanding the age and gender differences. The core aim of this study is to uncover new biomarkers that can contribute towards distinguishing between male and female according to the gene expression levels of skeletal muscle (SM) tissues. In our proposed multi-filter system (MFS), genes are first sorted using three different ranking techniques (t-test, Wilcoxon and ROC). Later, important genes are acquired using majority voting based on the principle that combining multiple models can improve the generalization of the system. Experiments were conducted on Micro Array gene expression dataset and results have indicated a significant increase in classification accuracy when compared with existing system.

KEYWORDS

Multi-Filter System, Filter Techniques, Micro Array Gene Expression, Skeletal muscle

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 237–244, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60619

1. INTRODUCTION

Sexual dimorphism of skeletal muscle can occur due to age [1] and many of these age-related changes in skeletal muscle appear to be influenced by gender [2], [17], [18]. For example, the muscle mass of men is larger than that of women, especially for type II fibers, while the type I muscle fibers proportion of oxidative is higher in women [3]. Welle et al. reported that the muscle mass of men is larger than that of women [1], [11], [12], due to the higher level of testosterone and the anabolic effect of testosterone is well known. However, previous studies have failed to identify which genes are responsible for anabolic effects. The molecular biases related to gender difference are still fuzzy [1]; 50% of the cell mass of the human body is muscle, so skeletal muscle is considered an important issue. There are several changes in skeletal muscle related to age that seem to be influenced by gender [4]. These changes in gene expression could be responsible for the decline in muscle function [5]. In relation to sex, despite the fact that there are a higher number of genes in expression related to gender difference, very few genes can help to interpret the gender difference issue [3]. For the profiles of men and women, there are few comparisons of broad gene expression that have been carried out [5].

Janssen et al [13] reported that the reduction of skeletal muscle (SM) mass related to age starts in the third decade. This decrease starts to appear in the lower body SM. To find differences between men and women, they used t-test, pearson correlation and multiple regression to determine the relationship between age and skeletal muscle. Dongmei et al [2] used basic statistical analysis to make a comparison between males and females in each set of age using gene expression profiles from skeletal muscle tissue. They identified important sex and age related gene functional groups using intensity-based Bayesian moderated t-test and logistic regression. This was the first study that offers global proof for the occurrence of extensive sex changes in the aging process of human skeletal muscle. Although the study showed interesting results, but they had used genes belonging to X and Y chromosomes, which can easily discriminate genders. Experiments were conducted using 3 groups namely older women versus old men, young women versus older women, and young men versus older men. But the main problem with their study is that important genes are identified using whole training data. This can lead to poor generalization because one of the fundamental goal of machine learning is to generalize beyond the samples in the training data.

The main aim of this paper is to extend the work reported by Dongmei et al [2] by identifying important genes with good generalization ability. In our proposed approach which is basically inspired from ensemble of feature ranking methods for data intensive application [16], genes are first sorted using three different ranking techniques (t-test, Wilcoxon and ROC). Later, important genes are acquired using majority voting based on the principle that combining multiple models can improve the generalization of the system. The scope of this paper is the selection of the most reliable genes and the evaluation of classification power of selected genes. Experiments were conducted on Micro Array gene expression dataset and results have indicated a significant increase in classification accuracy when compared with the genes obtained by the system in [2]. Our proposed technique is able to identify differentially-expressed genes for the following three case studies in relation to age and gender differences

- Young Women versus Old Women
- Young Men versus Old Men
- Old Men versus Old Women

This paper is organized as follows. Section II describes material and the proposed method followed by results and discussion in Section III. Section IV concludes the paper.

2. MATERIAL AND PROPOSED METHOD

A. Micro array gene expression data set

In this study, the dataset contains a microarray dataset of gene expression of skeletal muscle arm tissue. Dataset is publicly available in the Gene Expression Omnibus (GEO) database [2]. The subjects comprise 22 healthy males and females of various ages, in which 7 males & 7 females are young (20-29 years old), and 4 males & 4 females are old (61-81 years old). The whole Ribonucleic Acid (RNA) was extracted and gene expression profiling was implemented utilising Affymetrix human genome U133 Plus 2chip. As in [2], this data set is divided into three cases, first case involve 11 females (7 young and 4 old), second case consists of 11 males (7 young and 4 old) and the last case contains 8 samples (4 old men and 4 old women).

B. Genes subset selection using Feature ranking techniques

Bioinformatics data have extremely high dimensionality. The above dataset consists of around 55,000 genes with only 22 samples. This is considered a significant challenge to machine learning methods. This means that there are a large number of features than samples. To address this problem, it is important to select a small relevant features subset to reduce processing time and to avoid over fitting [6]. The possible solution is the feature ranking methods. In this study, three different filter methods are investigated and are shown below

- **T-test**: a statistical hypothesis where the statistic follows a Student distribution [9]. It is usually used to evaluate if the averages of two classes are not statistically similar by computing the variability and difference between two classes.

- **Entropy**: is normally used for high dimensional data to select the suitable number of features using the principle of Entropy.

- **ROC**: offers an active method to characterize the classifier sensitivity versus specificity.

C. Classification

Selected subset of genes are tested for its generalisation power using supervised classification. k-nearest neighbor (kNN) classifier (k=1,3) is used to used to evaluate the system performance. The leave-one-out cross validation (LOOCV) technique is used for evaluation.

Table 1 : Majority Voting to Select Important Genes.

	Top Ranked Genes
t-test	1, 4, 6, 9, 10
ROC	1, 2, 5, 9, 10
Entropy	3, 4, 5, 9, 10
Majority Voting	1, 4, 5, 9, 10

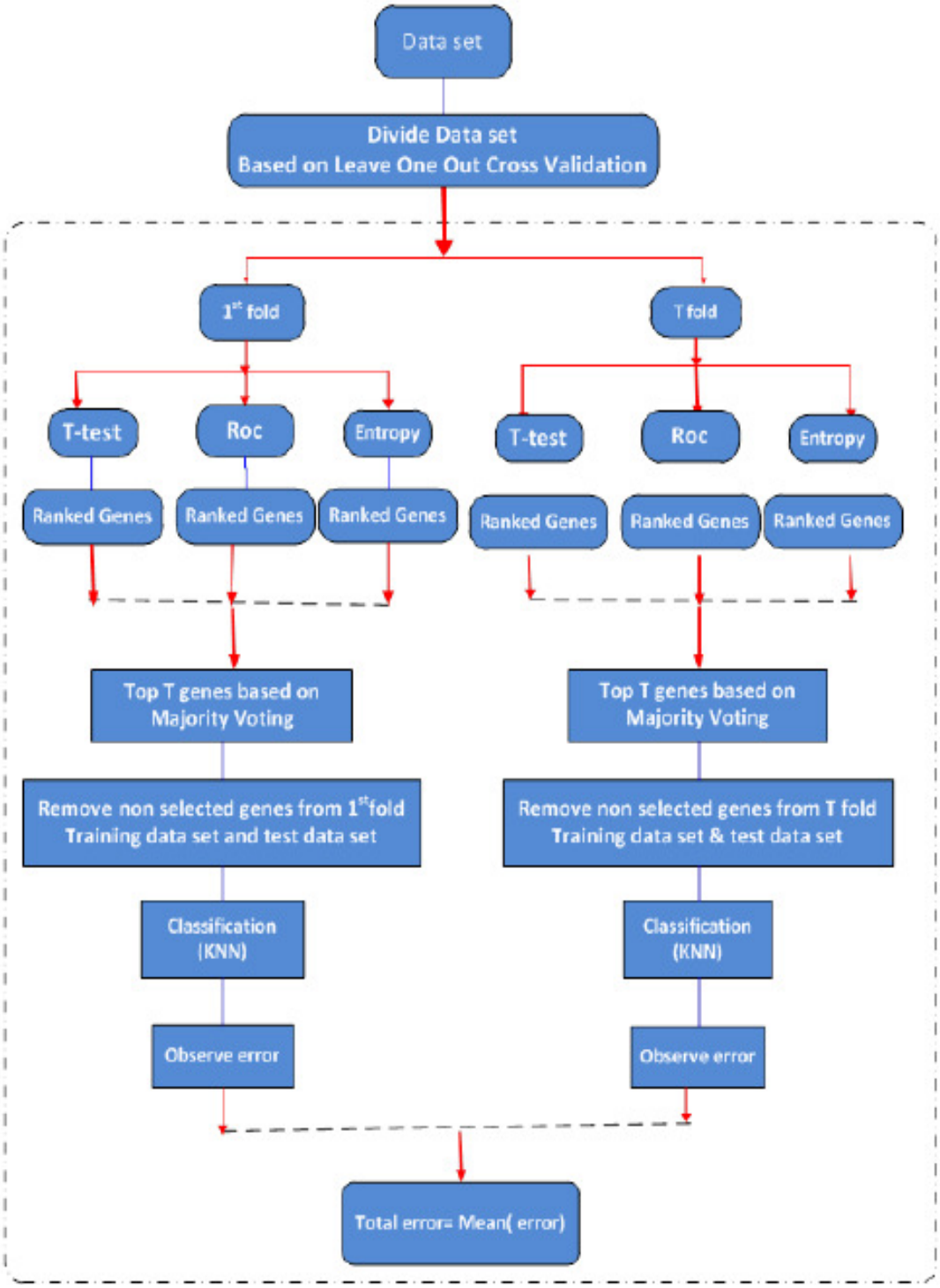


Fig. 1. Proposed Multi-Filter System (MFS).

1) *k-nearest neighbor*: The main objective of *k*-nearest neighbor (*k*-NN) classifier is to discover set of *k* objects in the training set that are similar to the objects in the test group [14]

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (a_{x_i} - a_{x_j})^2} \quad (1)$$

where *a* is the feature vector of x^{th} sample.

D. Proposed System

Figure 1 shows the framework of the proposed system which is inspired from the fact that combining multiple models can improve the generalization of the system. We first divided the data set using leave-one-out-cross validation into T folds. In other words, there are 20 folds for 20 samples where each fold consists of 19 samples for training and one sample for testing. For each fold multi filter system (MFS) is applied, which includes three different rank feature filters T-test, ROC and Entropy. Each filter ranking technique is responsible to sort genes according to criteria specified in the filter ranking methods. From these sorted genes, N unique subset of genes are obtained based on majority voting. This is clearly depicted using Table I. Lets assume that there are total 10 genes and the objective is to select top 5 genes. Genes 9, and 10 are selected by all feature ranking techniques so these are most important genes. Genes 1, 4, 5 are selected twice and thus are also considered as important genes by the system. It should be noted that due to majority voting, genes 2, 3 and 6 are not selected by the system. Later, kNN is applied on the new subset of genes in order to check the predictive performance.

3. RESULTS AND DISCUSSION

In this section, we will evaluate the performance of the multi-filter system (MFS). The proposed system is also compared with the system presented in [2], in which 75 genes are identified for three categories (male young versus male old, female young versus female old and male old versus female old) from total of 54623 genes. In order to have a fair comparison, the same number of genes are selected from MFS and compared with the genes identified in [2]. The evaluation metrics used in this study are: Classification accuracy, Sensitivity and Specificity.

A. Case Study 1: Young Men versus Old Men

This case study consists of 11 male samples (7 young and 4 old). Table II shows the performance of MFS when compared with the genes identified by Liu et al [2]. It is observed that the best performance is obtained using 3NN classifier which is 90.9% while genes obtained by [2] only able to achieve 81.8%. This improvement is mainly due to high specificity. Further analysis has revealed that out of 75 genes, only 9 genes are common in both systems. Some new genes are identified, that can play an important role in age differences of young and old males. Some of the new genes are shown in Table III along with 9 genes that are selected by both systems. These new genes can be very useful for biologist in order to identify the differences between young and old males.

Figure 2 shows the performance of the system by varying the number of genes. It is observed that the best performance is obtained by using 10 or 20 genes and afterwards, there is a 10% drop in performance. This may be due to selection of some genes that can degrade the performance of the system. Future work aims to investigate wrapper techniques to identify these genes.

Table 2. Young Men Versus Old Men

Classifier	Classification Accuracy		Sensitivity		Specificity	
	MFS	[2]	MFS	[2]	MFS	[2]
1NN	0.818	0.636	0.714	0.571	1.000	0.750
3NN	0.909	0.818	0.857	0.857	1.000	0.750

Table 3. Young Men Versus Old Men. New Genes Selected by the Proposed System. Common Genes Selected by Proposed System and System by [2].

New genes selected by MFS	Common Genes
Caveolin 3 (CAV3)	Toll-like receptor 4 (TLR4)
Eukaryotic translation elongation (EEF1B2)	UDP-GlcNAc:betaGal (B3GNT6)
FBR-MuSV ubiquitously expressed (FAU)	TGF-beta activated kinase 1 (TAB3)
RNA binding motif protein 15 (RBM15)	Myozenin 3 (MYOZ3)
Ribosomal protein L4 (RPL4)	Olfactory Receptor (OR5P3)
Cytochrome c-1 (CYC1)	Thioesterase superfamily member 4 (THEM4)
Mitochondrial ribosomal protein S30 (MRPS30)	RAN binding protein 3-like (RANBP3L)
Pyruvate dehydrogenase kinase, isozyme 2 (PDK2)	Fc receptor-like 3 (FCRL3)
Phosphoglycerate mutase 2 muscle (PGAM2)	Rhomboid, veinlet-like 3 Drosophila (RHBDL3)

B. Case Study 2: Old Men versus Old Women

This case study consists of 8 adults (4 old men versus 4 old women). Table IV shows the performance of MFS when compared with the genes identified by Liu et al [2]. It is observed that genes selected using MFS have classification accuracy of 100% using both 1NN and 3NN with high Sensitivity and Specificity.

C. Case Study 3: Young Female versus Old Female

This case study consists of 11 female samples (7 young and 4 old). Table V shows the performance of MFS when compared with the genes identified by Liu et al [2]. Again, the best performance is obtained using 1NN classifier which is 91%. While genes identified by [2] are only able to achieve 72.2% which indicates the important improved generalisation ability of the proposed system. We argue that improvement in performance is mainly due to high Specificity as Sensitivity which is same in the both systems.

4. CONCLUSION

In this study, muti-filter system (MFS) is proposed to identify important genes for Males and Females using skeletal muscle. Genes are first sorted using three different ranking techniques (t-test, Wilcoxon and ROC). The proposed system is evaluated on publicly available microarray dataset of gene expression of skeletal muscle arm tissue. Later, important genes are acquired using majority voting based on the principle that combining multiple models can improve the generalization of the system. The results have indicated that the classification performance achieved by the proposed system yields the best classification performance when compared with similar number of genes identified in previous study [2]. Future work aims to improve the performance by identifying more important genes through Wrapper Feature Ranking techniques rather than filter based feature ranking techniques.

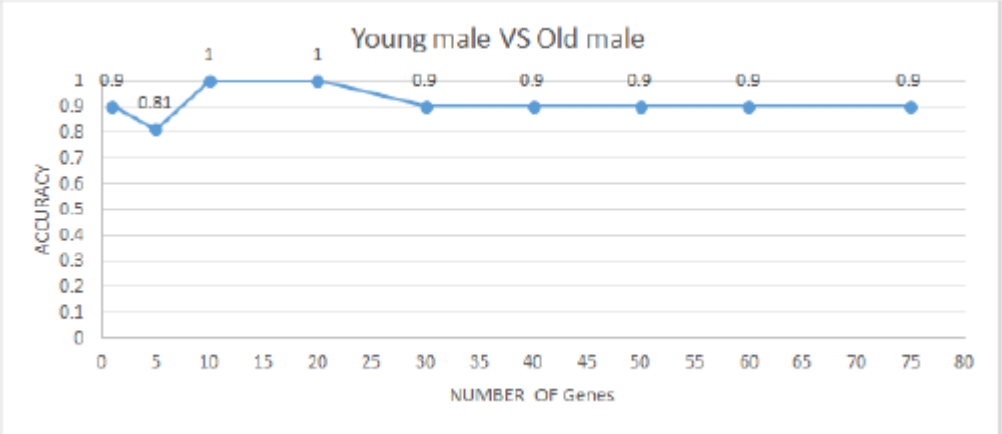


Fig. 2. Graph showing performance of the system using varying number of genes.

Table 4. Old Men Versus Old Women

Classifier	Classification Accuracy		Sensitivity		Specificity	
	MFS	[2]	MFS	[2]	MFS	[2]
1NN	**1.000**	0.750	1.000	0.500	1.000	1.000
3NN	**1.000**	0.375	1.000	0.500	1.000	1.000

Table 5. Young Female Versus Old Female

Classifier	Classification Accuracy		Sensitivity		Specificity	
	MFS	[2]	MFS	[2]	MFS	[2]
1NN	**0.909**	0.454	0.857	0.571	**1.000**	0.025
3NN	0.722	0.722	0.857	0.857	0.500	0.500

REFERENCES

[1] S. Welle, R. Tawil, and C. A. Thornton, "Sex-related differences in gene expression in human skeletal muscle", PLoS One, vol. 3, no. 1, pp. e1385-e1385, 2008

[2] D. Liu, M. A. Sartor, G. A. Nader, E. E. Pistilli, L. Tanton, C. Lilly, et al., "Microarray analysis reveals novel features of the muscle aging process in men and women", Biological Sciences, vol. 68(9), pp. 1035–1044, 2013

[3] D. D. Liu, M. A. Sartor, G. A. Nader, L. Gutmann, M. K. Treutelaar, E. E. Pistilli, H. B. IglayReger, C. F. Burant, E. P. Hoffman, and P. M. Gordon, "Skeletal muscle gene expression in response to resistance exercise: sex specific regulation", BMC Genomics, vol. 11, no. 1, pp. 659, 2010.

[4] G. Sifakis, I. Valavanis, O. Papadodima, and A. A. Chatziioannou, "Identifying Gender Independent Biomarkers Responsible for human Muscle Aging Using Microarray Data", Bioinformatics and Bioengineering (BIBE), pp. 1-5, 2013

[5] S. M. Roth, R. E. Ferrell, D. G. Peters, E. J. Metter, B. F. Hurley, and M. A. Rogers, "Influence of age, sex, and strength training on human muscle gene expression determined by microarray", Physiological genomics, vol. 10, pp. 181-190, 2002.

[6] Y. Saeys, I. a. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics", Bioinformatics, vol. 23, pp. 2507-2517, 2007.

[7] Y. Su, T. M. Murali, V. Pavlovic, M. Schaffer, and S. Kasif, "RankGene: identification of diagnostic genes based on expression data", Bioinformatics, vol. 19, pp. 1578-1579, 2003

[8] K. Murphy. "Machine learning: a probabilistic perspective". Cambridge MA: MIT Press, 2012.

[9] N. Thouleimat, D. Hernandez-Lobato, and P. Dupont, "Variance Estimators for t-Test Ranking Influence the Stability and Predictive Performance of Microarray Gene Signatures", European Conference on Computational Biology, 2010.

[10] S. Sahan, K. Polata, H. Kodazb, and S. Gne, "Anewhybrid method based on fuzzy-artificial immune system and k-nn algorithm for breast cancer diagnosis", Computers in Biology and Medicine, vol. 37, pp. 415-423, 2007.

[11] M. Visser, M. Pahor, F. Tylavsky, S. B. Kritchevsky, J. A. Cauley, A. B. Newman, B. A. Blunt, and T. B. Harris, "One-and two-year change in body composition as measured by DXA in a population-based cohort of older men and women", Journal of applied physiology, vol. 94, pp. 2368-2374, 2003.

[12] V. A. Hughes, W. R. Frontera, R. Roubenoff, W. J. Evans, and M. A. F. Singh, "Longitudinal changes in body composition in older men and women: role of body weight change and physical activity", The American journal of clinical nutrition, pp. 473-481, 2002

[13] I. Janssen, S. B, Heymsfield, Z. Wang, and R. Ross, "Skeletal muscle mass and distribution in 468 men and women aged 1888 yr", Journal of applied physiology, vol. 89.1 pp. 81-88, 2000.

[14] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, and S. Y. Philip, "Top 10 algorithms in data mining", Knowledge and information systems, vol. 14, pp. 1-37, 2008.

[15] A. C. Haury, P. Gestraud, and J.-P. Vert, "The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures", PloS one, vol. 6, p. e28210, 2011.

[16] W. Altidor, T. M. Khoshgoftaar and J V Hulse and A. Napolitano, "Ensemble Feature Ranking Methods for Data Intensive Computing Applications", Handbook of Data Intensive Computing, pp 349-376, 2011

[17] A. Y. Guo, K. S. LeunG, P. M. F. Siu, J. H. Qin, S. K. H. Chow, L. Qin, C. Y. Li, and W. H. Cheung, "Muscle mass, structural and functional investigations of senescence-accelerated mouse P8 (SAMP8)", Experimental Animals, vol. 64, p. 425, 2015.

[18] R. R. Kalyani, M. Corriere, and L. Ferrucci, "Age-related and disease-related muscle loss: the effect of diabetes, obesity, and other diseases", The Lancet Diabetes & Endocrinology, vol. 2, pp. 819-829, 2014.

VULNERABILITIES OF THE SSL/TLS PROTOCOL

Jelena Ćurguz

Department of IT development and services,
Post Office, Banja Luka, BiH
jelena.curguz@postesrpske.com

ABSTRACT

This paper analyzes vulnerabilities of the SSL/TLS Handshake protocol, which is responsible for authentication of the parties in the communication and negotiation of security parameters that will be used to protect confidentiality and integrity of the data. It will be analyzed the attacks against the implementation of Handshake protocol, as well as the attacks against the other elements necessary to SSL/TLS protocol to discover security flaws that were exploited, modes of attack, the potential consequences, but also studying methods of defense. All versions of the protocol are going to be the subject of the research but emphasis will be placed on the critical attack that the most endanger the safety of data. The goal of the research is to point out the danger of existence of at least vulnerability in the SSL/TLS protocol, which can be exploited and endanger the safety of the data that should be protected.

KEYWORDS

SSL/TLS Protocols, Handshake Protocols, Attacks,

1. INTRODUCTION

Internet today has become of great importance for the economy, education, business, and almost all other aspects of society, which is becoming an irreplaceable tool for work and for getting of the necessary information. Most companies organize their business via the Internet, all business communications, distribution, purchase, sale, marketing and servicing of products are made via the Internet. Environment such as e-banking, e-commerce, e-business and other offer many benefits to its users. The simplest way to perform a financial transaction is over the Internet. Use of cloud environments is becoming more popular, the possibilities it offers to its customers are very useful: access to data from anywhere, from any device, at any time.

The popularity of Internet is constantly increasing but carries with it certain security risks. A lot of data which are transmitted over the Internet infrastructure contain confidential and sensitive information (credit card number, user credentials, personal data,...) which require protection. Most corporations give their trust to SSL/TLS (Secure Sockets Layer/Transport Layer Security) protocol for data protection, which is also the most common way to protect data. However, because of its frequent use and role which it has for the protection of highly sensitive data, it is very attractive to detect and exploit security vulnerabilities.

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 245–256, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60620

2. SSL/TLS PROTOCOL

2.1. Introduction to SSL/TLS

SSL (Secure Sockets Layer), later called TLS (Transport Layer Security) is a cryptographic protocol designed to ensure the security of data transmitted over the Internet. Developed by the Netscape company in 1994 and in 1996 the company issued the latest version of this protocol called SSL3.0. Further development and release of the protocol took over the IETF organization but later named TLS. The protocol is used to protect the data on the transport layer.

It is located between the transport and application layer in the ISO/OSI reference model and provides security services for any application-based protocols, such as HTTP, FTP, LDAP, POP3,... It is used in client/server environment and provides following features for parties in communication:

- authentication,
- confidentiality
- integrity

2.2. Structure of the SSL/TLS protocol

SSL/TLS protocol consists of two layers and several protocols. The lower layer located next to the transport lavel in the OSI/ISO reference model consists of SSL/TLS Record protocol. The higher layer located immediately above the Record protocol consists of the SSL/TLS Handshaking protocols: Handshake protocol, ChangeCipherSpec protocol and Alert protocol. (Figure 1.)

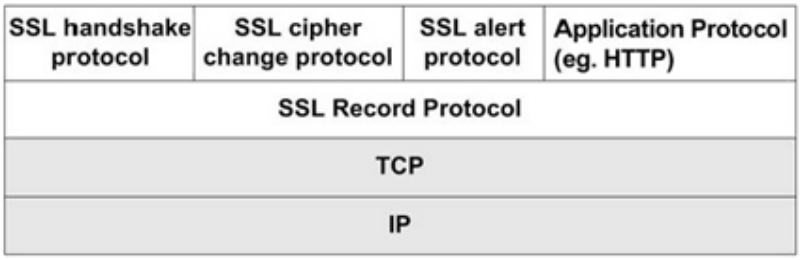


Figure 1. SSL/TLS protocol

2.2.1 Record protocol

The Record protocol is responsible for the transfer of blocks of data between the two sides in communication. It takes messages from application level of the OSI/ISO reference models, divides them into manageable blocks, optionally compresses, applies MAC, encrypts and transmits results. It uses security parameters negotiated during handshake phase.

2.2.2 Handshake protocol

The Handshake protocol is the core protocol of SSL/TLS responsible for authentication of each party of the communication and negotiation of security parameters to be used for exchange of encrypted data.

2.2.3. ChangeCipherSpec protocol

The ChangeCipherSpec protocol is used to notify both parties in the communication to upgrade the status of the session to negotiated parameters and move on to secure communication.

2.2.4. Alert protocol

The Alert protocol is used for the notification of errors that occur in communication between the two sides, i.e.: when the connection is closed, when the message can not be decrypted, etc..
During the handshake phase all cryptographic primitives responsible for connection protection are established. Communication between client and server during handshake phase is done with the messages with predefined forms. One example of exchanged messages between client and server during the handshake phase can be seen in Figure 2. The messages are:

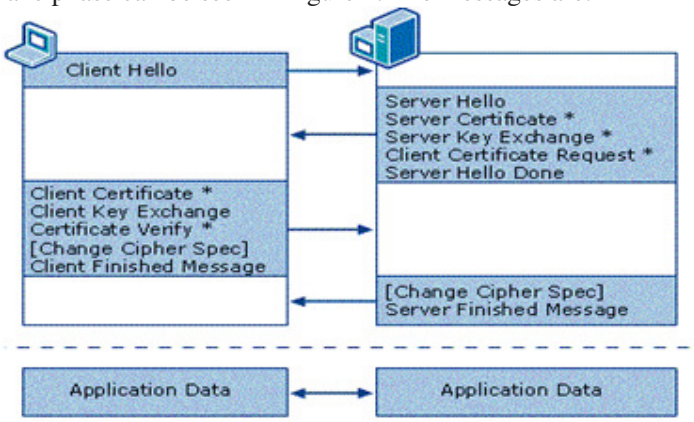


Figure 2. Messages of the Handshake protocol

1. ClientHello is type of message where the client notifies the server of the security parameters (protocol version, client random, session ID, cipher suite, compression method) which he supports and wants to use.

2. ServerHello is a message where the server notifies the client of the security parameters that will be used (protocol version, server random, cipher suite, session ID, compression method);

3. Certificate is a message which transmitted certificate for server and optional message for transmitted client's certificate.

4. ServerKeyExchange message is following ServerHello message if using anonymous negotiation or server Certificate message if there are not enough informations for the client in order to exchange premaster secret to the server. Optional message.

5. CertificateRequest message is a message where the server requires certificate from the client. Optional message.

6. ServerHelloDone message is a message where the server notifies that all requests are sent to the client in order to establish a communication.

7. ClientKeyExchange message contains generated premaster key encrypted with server's public key. Based on the premaster secret, later will be generated a master secret and based on the master secret will be generated all other keys in order to encrypt the traffic.

8. CertificateVerify message is a message where the client confirms that he has a private key corresponding to the public key from the certificate. This message is sent only if the client previously sent a client Certificate message. Optional message.

9. ChangeCipherSpec message is a type of message where the other party in communication is informed about the start of the use of agreed security settings. After these messages all the other messages that are exchanged are encrypted.
10. Finished message informs that all the steps of negotiations are done and that a secure communication is established.

When the process of negotiation of parameters and generating of necessary keys is finished, Record protocol takes the data from the application level, decomposes it into smaller blocks, optionally compresses it, applies the agreed hash function, encrypts with the agreed algorithm and then transport it through the transport layer to the other side in communication.

3. ATTACKS ON THE SSL/TLS PROTOCOL

SSL/TLS protocol is a widespread used protocol for data protection. However, because of that, it is very interesting for discovering and exploiting security flaws that harm the integrity and security of data. This paper will briefly analyze some of the attacks on SSL/TLS Handshake protocol, as well as other elements required in the work protocols, to analyze security flaws that were exploited, modes of attack, the potential consequences, but also studying methods of defense. The attacks, which will be discussed are:

- Ciphersuite rollback attack
- Drop ChangeCipherSpec attack
- Version rollback attack
- Key Exchage Algorithm confusion
- Bleichenbacher attack on PKCS#1
- Attacking RSA-Based Sessions in SSL/TLS
- Timing attack
- Cross-protocol attack based on ECC key exchange
- New Bleichenbacher side channels and attacks
- SKIP-TLS attack
- FREAK atack
- Logjam attack

3.1. CipherSuite rollback attack

Cipher suite is a list of cryptographic algorithms that are proposed during the handshake phase between the client and server. List of proposed algorithms is traveling in clean text format as part of the initial ClientHello messages. It allows MITM attacker to intercept the message and replacing client's cipher suite with his cipher suite that supports weaker versions of algorithms or NULL-Cipher list, so communication continues to take place with weaker algorithms or algorithms for protection are not used at all [2]. The consequences of such an attack could be disastrous for the client: the attacker could imitate a valid user, could access the server, obtain user credentials and the like.

In version SSL3.0 this failure is resolved with the authentication of all handshake messages in the final Finished message, which contains MAC on handshake protocol messages, keyed by the

master secret, so the attack on cipher suite might be noticed at the end of the handshake phase and reject such a session [2].

3.2. Drop ChangeCipherSpec attack

Drop ChangeCipherSpec attack is weakness of SSL2.0 discovered by David Wagner and Bruce Schneier. The ChangeCipherSpec message is used to notify both parties in the communication to upgrade the status of the session to negotiated parameters in the handshake phase. When the initial handshake phase is completed, the client and server exchange ChangeCipherSpec message to signal the other side that in the future all communication will be done only with the agreed parameters. However, before one side send the ChangeCipherSpec message MITM attacker can send Finished message to other side, which furthermore would cause the start of communication without any changes and adoptions of agreed security parameters, or it could simply delete ChangeCipherSpec message, so the client and the server would never establish a communication [2].

The solution for this problem is to force both parties to ensure that a ChangeCipherSpec message is received before accepting the Finished message. RFC 2246 says "It is essential that a ChangeCipherSpec message is received between the other handshake messages and prior to the Finished message" [5].

3.3. Version rollback attack

Version rollback attack is a vulnerability of SSL 3.0. It is a type of attack where the attacker can lead the client and server side of the communication to use a lower version of the protocol SSL2.0 instead of one they should use and support SSL3.0. The attacker modifies ClientHello message SSL3.0, so it looks like ClientHello message SSL2.0. In that way it can use the weaknesses of protocol SSL2.0, like weaker cipher suite of proposed algorithms (i.e. a list with the DES algorithm, which further allows an attacker brute force attack and compromising of sensitive data).

Paul Kocher has designed a strategy to detect version rollback attacks so that clients that support SSL3.0 have built-in fixed redundancy in the RSA PKCS padding bytes in order to indicate that they support SSL3.0. Servers will refuse SSL2.0 connection if they notice that these bytes are present on the client side [2]. Unfortunately, this strategy is valid only for RSA key exchange but not for Diffie Hellman. SSL 3.0 and later TLS versions offer protection for version rollback attacks with the Finished message (ClientHello message contains the client_version field, which shows its highest supported version). Also in case RSA, the client generates a 48-byte premaster secret, and the version number in the premaster secret is the version offered by the client in the Client Hello, not the version negotiated for the connection.

3.4. Key Exchange Algorithm confusion or Cross-protocol attack

Key Exchange Algorithm confusion or Cross-protocol attack is vulnerability of SSL3.0. Server can send to the client the temporary key parameters signed under its long-term certified signing key in ServerKeyExchange messages [2]. The problem is that the signature of the temporary key parameters does not include part of the field where it is specified which type of key is used, the RSA or Diffie Hellman, and thus created a basis for a confusion type of attack.

The attacker forced the server to use the Diffie-Hellman key exchange and client to use RSA key exchange. This leads to confusion where the client may interpret the Diffie-Hellman parameters

(p, g) as an exponent and module of RSA key. In the following example we can see how the attack works [2].

[ClientHello]
Client ->Attacker: SSL_RSA…
Attacker->Server: SSL_DHE_RSA…
[ServerHello]
Server->Attacker: SSL_DHE_RSA…
Attacker->Client: SSL_RSA…
[ServerKeyExchange]
Server->Attacker: $\{p,g,y\}K_s$
Attacker->Client: $\{p,g,y\}K_s$
[ClientKeyExchange]
Client->Attacker: $k^g \bmod p$
Attacker->Server: $g^x \bmod p$

$k^g \bmod p$, k is premaster secret. For successful attack the client intercept premaster key which will be encrypted with RSA key, or Diffie Hellman parameters *(g, p)* which attacker already knows. Furthermore, the attacker sends to server $g^x \bmod p$ whereby the server interprets premaster key as $g^{xy} \bmod p$. Attacker in future can intercept, read and change all the exchanged messages between the client and the server, act as a server to the client and as a client to the server.

Proposed solution for cross protocol attack was a new protocol extension indicating the new format of ServerKeyExchange message which includes explicit indicators of the entity (server), the type of key exchange algorithm, the handshake messages exchanged and the parameters of the key exchange [4].

3.5. Bleichenbacher attack on PKCS#1

Daniel Bleichenbacher in 1999 presented the attack where it is possible for a certain amount of time to decript premaster secret encrypted with server's RSA public key. Premaster secret encrypted with RSA algorithm is the value generated by the client and sent to the server (encrypted and formatted with the standard PKCS v1.5) within ClientKeyExchange messages.
To encrypt a message M, with the RSA public key in accordance with the standard PKCS v1.5 format of message must be

0x00 0x02 [non-zero bytes] 0x00 [M]

At the beginning of the message always comes *0x00 0x02* bytes, after padding bytes, *0x00,* and finally message *M*.

The attack is based on the chosen ciphertext attack. The core of Bleichenbacher's attack relies on an *oracle*: the attack works if there is some system, that for any chosen ciphertext *C*, indicates whether the corresponding plaintext has the correct format according to the standard PKCS#1 [8]. Scenario for an attack is the following:

- The attacker has access to a system (*oracle*) that for each selected encrypted data returns the value *true* or *false* depending on whether the decrypted message is or not in accordance with PKCS#1 v1.5 structure.

- The attacker eavesdropps the communication and wants to decrypt the encrypted information *C*. He knows that $C = M^e \bmod n$. He wants to calculate $M = C^d \bmod n$, or the premaster secret.

Attacker sends a large number of requests with a random value and requires from the system (*oracle*) to decrypt messages $C' = s^e C \bmod n$. System decrypt the message as $M' = (C')^d \bmod n$.(Figure 3)

Based on multiple responses M' which are in accordance with PKCS#1 v1.5 standard, attacker concludes about the possible values of M and decrypts the message as $M = M's^{-1} \bmod n$ [8].

If the oracle answers with "true", the attacker knows that messages M start with *0x00 0x02* and $2B < M < 3B - 1$, where $B = 2^{8(k-2)}$ and k bytes length of n. If the oracle responds with "false", the attacker increments s and repeats request to oracle. By iteratively choosing new values for s, querying the oracle, the attacker narrows down the interval which contains the original M value.

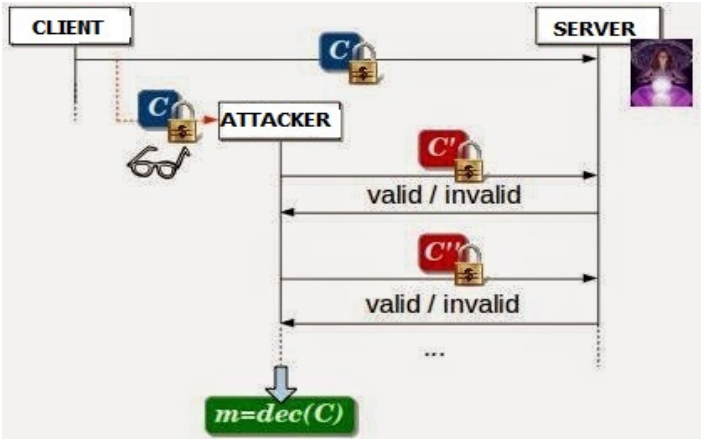


Figure 3. Bleichenbacher attack

Possible defense from the attack is that the server does not inform the client of irregular PSCS#1.5 format of message. The RFC 2246 says: "The best way to avoid vulnerability to this attack is to treat incorrectly formatted messages in a manner indistinguishable from correctly formatted RSA blocks. Thus, when it receives an incorrectly formatted RSA block, a server should generate a random 48-byte value and proceed using it as the premaster secret. Thus, the server will act identically whether the received RSA block is correctly encoded or not" [5].

3.6. Attacking RSA-Based Sessions in SSL/TLS

In 2003 the researchers Klima, Pokorny and Rosa presented "bad version oracle" attack. A countermeasure from the Bleichenbacher's attack from 1999 is to generate a random premaster secret and continue with the handshake phase until the verification and decryption of the Finished message fails, due to different key material.

Encrypted data of the ClientKeyExchange message in an RSA-based handshake includes also the major and minor version number of protocol offered by the client. Format of message is

0x00 0x02 [non-zero bytes] 0x00 [0x03 0x01 Random baytes]

where *[0x03 0x01 Random baytes]* is a message M or premaster secret and *0x03 0x01* means TLS1.0 or SSL3.1.

Many implementations checked for equality of the received protocol version contained in the ClientKeyExchange message to the one expected. This check requires a valid PKCS#1 v1.5 structure. Unfortunately, it has not been specified how such a check may be combined with the

countermeasure from the Bleichenbacher's attack and therefore creates the basis for build a "bad version oracle" attack. Form of attacks is an extended version of Bleichenbacher's attack.

$$O_{BadVersion}(x) = \begin{cases} true, & if\ version\ number\ is\ valid \\ false, & otherwise \end{cases}$$

In case of protocol version mismatch an Alert message was returned to the sender and sender knows that message is in accordance with PKCS#1 structure and starts with *0x00 0x02*.

The authors propose to keep generating premaster secret randomly if messages is not PKCS conforming. They propose to replace major and minor version number of protocol with the expected version number in either case (i.e. if message is or is not PKCS-conforming) [9]. Furthermore, RFC 5246 says "In any case, a TLS server MUST NOT generate an alert if processing an RSA-encrypted premaster secret message fails, or the version number is not as expected. Instead, it MUST continue the handshake with a randomly generated premaster secret"[10].

3.7. Timing attack

Timing attack is a form of side channel attack that exploits time to reveal the encrypted data. Attack exploits timing variants of cryptographic operations for different values of the input data. This attack computes the private key on the server by calculating the time difference between sending a specially made ClientKeyExcange messages and receiving Alert message that alerts the irregular premaster secret [3].

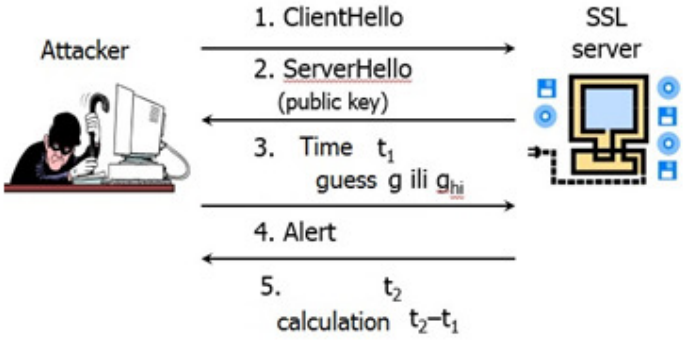


Figure 4. Timing attack

Encryption of message M with RSA algorithm is $C = M^e \bmod n$, and decryption is $M = C^d \bmod n$, where (e,n) is RSA public key and d is private RSA key. The goal is to find the private key d [6]. The attacker must have access to the target system that will count $C^d \bmod n$ for a few selected values of C. (Figure 4) With the precise count of time and with analyzing different time variants, with sending ClientKeyExcahge messages (different values of C) and receiving Alert messages, the attacker calculates the individual bits of the private key $d = d0d1\ldots$.

The most widely used method for defense from this attack is RSA blinding. Because the timing attacks expose information by measuring the amount of time required to perform mathematical operations, before decrypting the ciphertext C system first compute $X = r^e C \bmod n$, where r is a random value. For decrypt X system compute $X^d \bmod n = r^{ed}\ C^d \bmod n = r\ C^d \bmod n$. Then the output multiply by r^{-1} to obtain $M=C^d \bmod n$ which is the plaintext. Since a different r is used for each message, blinding prevents an attacker from computation time variant operations during decoding [11].

3.8. Cross-protocol attack based on ECC (Elliptic Curve Cryptography) key exchange

In 2012 serveral researchers discovered new cross protocol attack similar cross protocol attack from 2003. The main idea is the same as in the earlier attack, but now at the client side the attacker will negotiate ephemeral Diffie-Hellman key exchange, and on the server side ephemeral Elliptic Curve Diffie-Hellman key exchange. The attacker will intercept the TLS handshake messages between the server and the client and alter some of them that the client thinks that it is used ephemeral Diffie-Hellman key exchange and server ephemeral Elliptic Curve Diffie-Hellman key exchange. The goal of an attacker is to impersonates a server to the client and puts itself in between them, for the opportunity to reads, changes and forwards all the messages [13].
Proposed solution for cross protocol attack was a new protocol extension indicating the new format of ServerKeyExchange message which includes explicit indicators of the entity (server), the type of key exchange algorithm, the handshake messages exchanged and the parameters of the key exchange [4].

3.9. New Bleichenbacher side channels and attacks

In 2014 researchers presented a few new Bleichenbacher side channels and attack. They present four new Bleichenbacher's side channels and three successful Bleichenbacher attacks against the JSSE (SSL/TLS implementation) and against hardware security appliances using the Cavium NITROX SSL accelerator chip. These attacks are: error messages in JSSE, timing differences in JSSE, GnuTLS and OpenSSL implementations, internal exception in JSSE and unexpected timing behavior by hardware appliances [14].

For all these vulnerabilities researchers were found patches.

3.10. SKIP-TLS: Message Skipping Attacks on TLS

SKIP-TLS is a set of vulnerabilties found in popular open source TLS implementations (OpenSSL, GnuTLS, CyaSSL, JSSE,...) during 2015. Researchers tested popular open source implementations for state machine bugs and discover several new critical security vulnerabilities. They find that several TLS implementations incorrectly allow some handshake messages to be skipped even though they are required for current handshake [15].

Patches for these vulnerabilities are in development.

3.11. Freak (Factoring Attack on RSA-EXPORT Keys) attack

The attack discovered in March 2015 exploits the weaknesses of some implementations of SSL/TLS protocols that supports the "export-grade cryptography" [16]. Attacker by MITM attack proposes to server the use of RSA_EXPORT cipher suite that uses weaker RSA keys, i.e. keys of 512 bits or even less. Server accepting such a demand endangers the detection of weak RSA key and decrypion of the traffic. The vulnerabiliy was due to the previous law of the US government where it didn't allow to export strong cryptographic algorithms in other countries.

Defense against this attack is to forbid cryptography based on weak cryptographic algorithms created under the former law of the US government.

3.12. Logjam attack

The attack discovered in June 2015 is similar to Freak attack, because it is based on the already

well-known weaknesses of the TLS protocol that supports cryptography arising from the previous law of US government for the other countries, but this time was not directed at the RSA key exchange but on the Diffie-Hellman.

Diffie-Hellman key exchange is a widespread method of securely exchanging cryptographic keys. It is main method for the SSH and IPsec protocol and one of the options for TLS. The way to generate secret keys works as follows: Alice and Bob agree on a primary number p and a generator g. Alice sends $g^a \ mod \ p$ to Bob and Bob sends $g^b \ mod \ p$ to Alice, but each of them calculates the secret key $g^{ab} \ mod \ p$. The best technique for the attack on the Diffie-Hellman key exchange is based on the compromising the one of the private exponent *(a, b)* calculating the discreet log. The attacker who can find the discrete log x of $y = g^x \ mod \ p$ easily calculate the private key [17].

Previous law of the US government would not allow export of the algorithm in which the prime number was greater than 512 bits. Logjam attack is MITM attack in which an attacker can do downgrade TLS connection to the 512-bit "export-grade cryptography" connection. As we know, the client sends a list of proposed algorithms to server within ClientHello messages and based on that server chooses a list and indicate its choice in ServerHello message. Protocol supports several different variants of Diffie-Hellman key exchange: ephemeral, fixed and anonymous Diffie-Hellman key exchange.[1] The ephemeral Diffie-Hellman key exchange is mostly used.[1] At ephemeral Diffie-Hellman key exchange server is responsible for the selection of the parameters *(p, g),* and it calculates $g^b \ mod \ p$ and sends ServerKeyExchange message that contains the signature of the selected parameters (p, g, g^b) by signing key. The client verifies the signature and responds with ClientKeyExchange message that contains g^a. Both sides, on the basis of shared parameters, are calculating a master secret as $g^{ab} \ mod \ p$ and calculates a MAC of all exchanged handshake messages and exchanges them in the Finished message.[17] After calculating the encryption keys client and server may begin secure communication. The attacker, who is placed in the middle of communication, intercepts ClientHello message and removes all other lists of proposed algorithms and instead all DHE lists of algorithms puts DHE_EXPORT list that server accepts, if it could support it. Then it waits for ServerHello message where it also changes the DHE_EXPORT into DHE list and forwards ServerKeyExchange message. The client will interpret the DHE_EXPORT parameters (p, g, g^b) as valid DHE parameters selected by the server and will continue with the handshake phase. The attacker who can calculate secret *b,* from ClientKeyExchange messages, in the given time and based on that it can calculate the master secret and the encryption keys, so it can finish the handshake phase with the client in due time. After that he can continue reading and changing data with the client imitating the server [17].
Defense against this attack is to forbid cryptography based on weak cryptographic algorithms created under the former law of the US government.

4. CONCLUSION

Although, in this paper we didn't analyse all the attacks on SSL/TLS Handshake protocol, it can be concluded that in the long history of the SSL/TLS protocol, exactly this part of the SSL/TLS protocol has been exposed to various types of attacks. Negotiation of security parameters is the most critical part in work of the SSL/TLS protocol, whose compromising completely endangers the security of data transmitted to transport layer. Previous attacks on SSL/TLS protocol took the adventage of the fact that there was no authentication of the messages during handshake phase, no verification in the order of arrival of the handshake messages, used weakness of PKCS#1 v1.5, used some form of side channel attacks (time) and the like. Most of these attacks are implementation weaknesses of SSL/TLS protocol, while others are weaknesses in other elements used in work protocol. However, 20 years later researchers have been still finding some weaknesses in the protocol such as state machine bugs, support "export-grade cryptography",

Bleichenbacher side channels attacks and other.

The conclusion based on the our research is that the maximum attention was devoted to the entire mode and security of Handshake protocol and maximum effort was made to find ways of a defense from previous attacks. However, as any other sistem, so the SSL/TLS protocol can not be completely safe and there is always the danger of finding new vulnerabilities and their exploitation. Because of its large role in the protection of highly sensitive data from all attacks on SSL/TLS protocol, whether they are directed to the protocol itself or to some other elements, it is necessary to find a solution for the defense and to eliminate security flaw.

REFERENCES

[1] Rolf Oppliger, "SSL and TLS: Theory and Practice", 2009

[2] D. Wagner and B. Schneier. Analysis of the SSL 3.0 protocol. In Proceedings of the 2nd USENIX Workshop on Electronic Commerce, volume 2 of WOEC, pages 29–40. USENIX Association, 1996

[3] Christopher Meyer and Jorg Schwenk , "Lessons Learned From SSL/TLS Attacks",

[4] N. Mavrogiannopoulos "Preventing cross-protocol attacks in TLS protocol draft-mavrogiannopoulos-tls-server-key-exchage-00", 2012

[5] RFC 2246, "The TLS Protocol Version 1.0", 1999

[6] http://crypto.stackexchange.com/questions/12688/can-you-explain-bleichenbachers-cca-attack-on-pkcs1-v1-5

[7] Christopher Meyer, "20 Years of SSL/TLS Research An Analysis of the Internet's Security Foundation", 2014

[8] Bleichenbacher, D., "Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1," Proceedings of CRYPTO '98, Springer-Verlag, LNCS 1462, August 1998,

[9] Vlastimil Klíma, Ondrej Pokorný and Tomáš Rosa, "Attacking RSA-Based Sessions in SSL/TLS", 2003

[10] RFC 5246 The Transport Layer Security (TLS) Protocol Version 1.2

[11] http://www.cs.sjsu.edu/faculty/stamp/students/article.html#authorbio,"Timing Attacks on RSA", Wing.H.Wong

[12] https://en.wikipedia.org/wiki/Side-channel_attack

[13] Andrija Jakovljević, "Exploring cross-protocol attacks on the TLS protocol"

[14] Somorovsky, Eugen Weiss, Jörg Schwenk, Sebastian Schinzel, Erik Tews "Revisiting SSL/TLS Implementations: New Bleichenbacher Side Channels and Attacks", 2014

[15] Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, Cedric Fournety, Markulf Kohlweissy, Alfredo Pironti, Pierre-Yves Strubz, Jean Karim Zinzindohoue "Taming the Composite State Machines of TLS", 2015

[16] https://www.smacktls.com/, "Factoring RSA Export Keys"

[17] David Adrian, Karthikeyan Bhargavan,.. "Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice".

AUTHORS

Jelena Ćurguz has been employed in the Post of Republic of Srpska in Banja Luka, Bosnia and Herzegovina since 2003. She has been a Head of the Service for system support since 2008. She acquired extensive experience in the maintenance of systems and services in the Linux platform, Internet services, Informix database etc. She owns certificates in the field of Red Hat Linux. She is attending postgraduate studies at the Faculty of Electrical Engineering in Banja Luka.

SEMANTIC ANALYSIS OVER LESSONS LEARNED CONTAINED IN SOCIAL NETWORKS FOR GENERATING ORGANIZATIONAL MEMORY IN CENTERS R&D

Marco Javier Suárez Barón

PhD in Strategic and Technology Management,
UNITEC/FODESEP, Bogotá, Colombia
marcojaviersuarez@gmail.com

ABSTRACT

This paper shows the construction of an organizational memory metamodel focused on R&D centers. The metamodel uses lessons learned extracted from corporative social networks; the metamodel aims to promote learning and management of organizational knowledge at these types of organizations. The analysis is applied initially from lessons learned on topics of R&D in Spanish language. The metamodel use natural languages processing together with ontologies for analyze the semantic and lexical the each lesson learned. The final result involves a knowledge base integrated by RDF files interrogated by SPARQL queries.

KEYWORDS

Knowledge management, Technological Innovation Management, Lessons Learned, Ontology, Metamodel

1. INTRODUCTION

In Colombia, and specifically in the capital Bogotá, the R&D centers have become a key element in the scaffolding of science and technology [1]. R&D centers are principally affiliated with universities and have been primarily conceived of as technology-based companies. According to the science and technology observatory [2], they are currently the source of human capital formation in terms of research and contribute to knowledge by solving problems with technology. For [3] knowledge generation and storage activities have become a necessity in the creation of competitive advantages in modern organizations. Under this light, this article proposes a model that determines the relevance of the knowledge and experience that circulate in these science and technology organizations as a key factor in establishing organizational learning strategies from lessons learned. The aim of this model is provide strategic elements to support organizational learning through lessons learned extracted from social networks; the model apply natural

Jan Zizka et al. (Eds) : CCSEIT, AIAP, DMDB, MoWiN, CoSIT, CRIS, SIGL, ICBB, CNSA-2016
pp. 257–267, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60621

language processing(NLP) and ontologies giving support to knowledge management in R&D centers in Colombia.

2. BACKGROUND

Our literature review shows that innovation, development and research centers, known as R+D centers, have had a key role at all levels in modern society, to promote and encourage creativity around the world. A variety of innovative and creative ideas are closely related to investigation projects that have provided a solution to current problems for humanity. These solutions respond to new technological necessities, through feasible, viable innovative and creativity methods [4].
First of all, creativity can be directed to the generation of new and useful ideas, whereas innovation is regarded as the process of transforming the best ideas into real products. [5], from New Delhi University, claim that creativity is an individual activity, whereas innovation is a team work. This study concludes that all innovation processes start with a necessity that leads to a creative idea.

Additionally, in R+D centers, social capital has risen as a proper work frame to explain exchange mechanisms for organizational knowledge [6]. Lessons Learned, as well, are considered a type of knowledge for organizational learning which comes from experience, as it is claimed by [7]. In consequence, for knowledge transference to satisfy organization necessities, it is essential that lessons learned would be presented at the precise context and time in processes, so that they could be adapted to a learning process, as it was found by [8].

[9] Have stated that varied knowledge can move around from individual to individual. This research, carried out in collaboration with the Florida University in USA and the University of Bergen in Norway, shows that this knowledge reusing is not an easy task, but if it is extracted and retrieved in an efficient way, it could become a string strategy for organizational learning. Certainly, it is clear that the necessity to plan, test and measure new models directed to specific knowledge reusing and the supply of a latent space for this aim could be found in R+D groups.
Finally, if it is possible to organize and articulate any type of knowledge stored in any organization or collective's repository, including those belonging to a research group, then that knowledge, originating from varied sources, could be integrated as part of a single source [10]. Integration is an important activity in knowledge reuse [11], because it makes easier to track different pieces of knowledge that might be retrieved later, enhancing in that way, the process of knowledge reuse.

3. METHODOLOGY

According to [12], organizational learning models lead to applying meta-learning usage comprising learning cycles, mentioned above. This meta-learning concept contributes directly to innovation, enhancing all organizational learning. In [13] is described how organizations could learn from their own innovation and development projects, as well as those adopted by other organizations.[14] states that models that are developed for organizational learning lack integration choices that could define permanent metamodels for organizational learning, especially those concerning technology surveillance issues in R+D groups. That way, this knowledge might be shared, assembled and reused across a variety of organizational levels in a consistent way, using methods such as data ontologies [15].

This investigation has a mixed research approach including both qualitative and quantitative features, as a result of the usage of both approaches at different stages along the research process. Trends of R+D issues in lessons learned were defined using a qualitative approach; they were determined through text structures contained in social networks with the aim of establishing a metamodel based on knowledge management and organizational learning for researchers in R+D groups. On the other hand, determining statistical studies about behavioral background in relevant issues for R+D was made using a quantitative approach. It was focused on providing support to R+D groups in their decision taking and planning processes. This investigation has a correlational scope, since it was directed towards examining and analyzing technological and social variables behavior.

When improvement opportunities are considered, this type of model provides planning tools, since it makes possible to generate diagnosis using the information stored in social networks and its integration to other sources, such as documents and repositories, as it is stated by [16]. Also, it is possible to design strategies that could address R+D data for forecasts fulfillment. Finally, according [17] an organization might be analyzed and cleansed through the use of information ontologies, so that it could be structured as learning metamodel, becoming a valuable tool for organizational planning and learning.

The chart displayed in Figure 1 shows the framework for integration of Metamodel with organizational learning architecture. Here the lessons learned in the top layer, are supported by tacit and explicit knowledge sources that circulate into corporate social networks. That knowledge collects information originating from workplace issues and people, respectively. They, in turn, receive information from the archive database and provide, simultaneously, data for integration to the organizational memory.

Within this proposed framework, we can see the application of the basic concepts for facilitating personal knowledge management (which in this article we have called “lessons learned"), integrated with the corporate social networks of each research center (tacit knowledge –> explicit knowledge). Therefore, profiling each person or group of people is imperative for the creation of knowledge. A computer application is developed to perform this task. It allows the extraction of information in real time from lessons learned, in ranges or time periods defined by the research community. Also the framework allows us to standardize concepts, practices, and criteria in order to apply the proposed metamodel, and it will serve as a reference to confront and solve new problems of a similar nature.Additionally, the framework aims to promote new ways of capturing knowledge, using sources such as lessons learned that circulate in social networks.

The model uses an organizational learning structure for knowledge management. It can also be observed in Figure 1, the model includes an analysis of six shared knowledge sources: people, processes, documents, issues and, tacit and explicit knowledge. In the same way, this model extracts from an ontology, denominated here as R+D ontology, vocabulary from the R+D field, its interrelations, concepts, and metadata, applied to the integration of retrieved knowledge. Natural Language Processing (NLP) techniques are used for metadata, so that they could be used in a correlational technique analysis. Finally, it can be observed the flow and relations existent between the organizational elements or entities that are part of the metamodel; in other words, the data-information-knowledge progression.

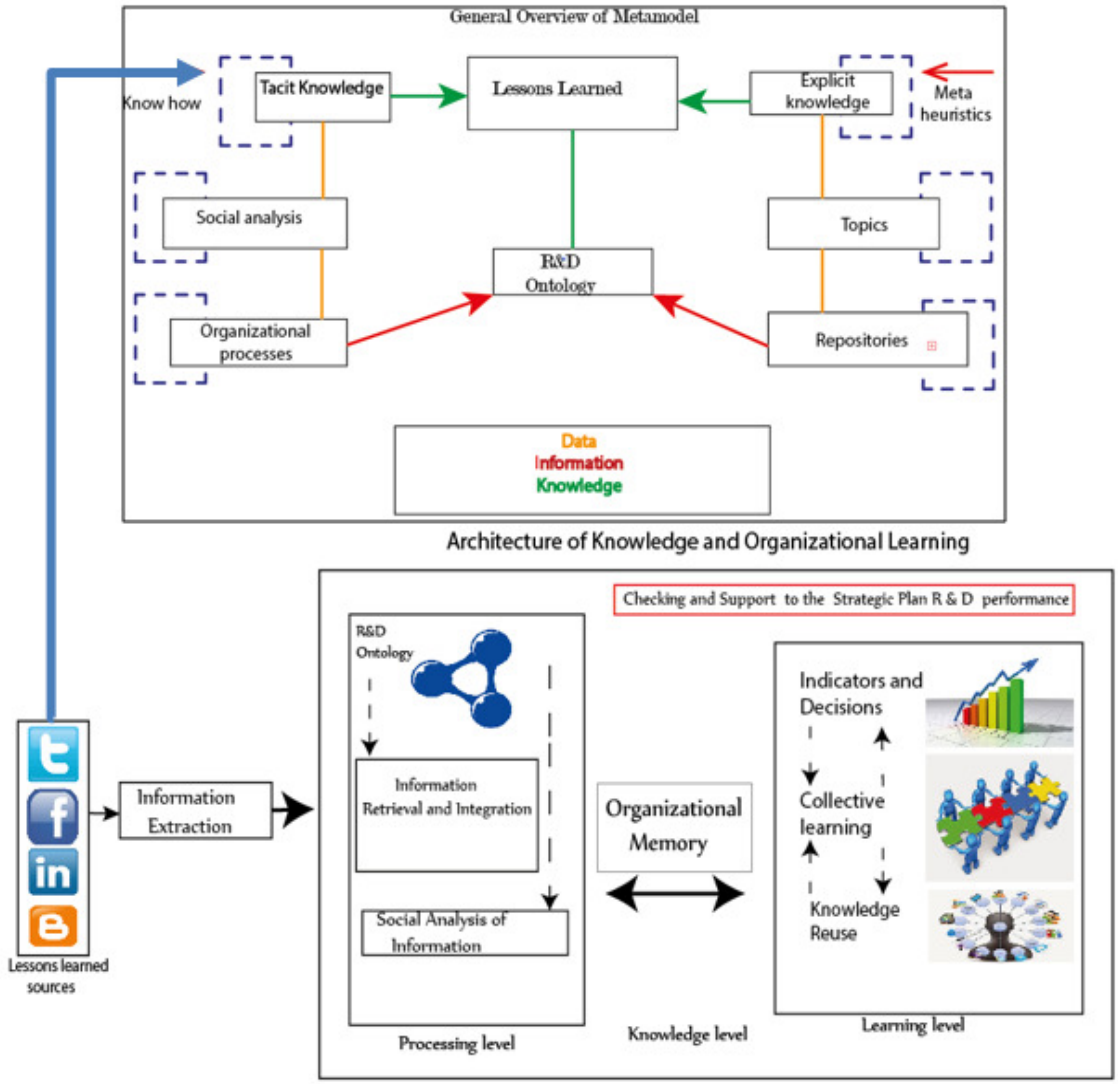


Figure 1: Global Architecture for the Knowledge Management Model

4. RESULTS AND DISCUSSION

R+D ontology is a functional component in our framework; it provides a semantic solution because it defines the linguistic and hierarchic relations among specialized concepts and characteristics in the R+D field, which are based on terminology from strategic planning on R+D. It also provides essential metadata for a declaratory representation of knowledge that can be communicated among people, processes and technology. Finally, it provides a formal definition for concepts that are agreed on, ensuring the correct interpretation of shared knowledge and providing a well-defined common vocabulary for information and knowledge exchange purposes in this field.

The *METHONTOLOGY* methodology was used for developing R+D ontology. As it was stated by [18] this method provides guidelines to specify ontologies at a certain knowledge level, specifically providing techniques to determine conceptualization. The OWL ontology

programming language was also used for designing and building R+D ontology; it was executed under the Protégé 4.3 platform. As it can be seen in Figure 2, the categories diagram design was elaborated taking into account four main entities,: Thing, R+D, Innovation and, Development and Innovation; these categories are merged to generate the complete structure of the ontology.

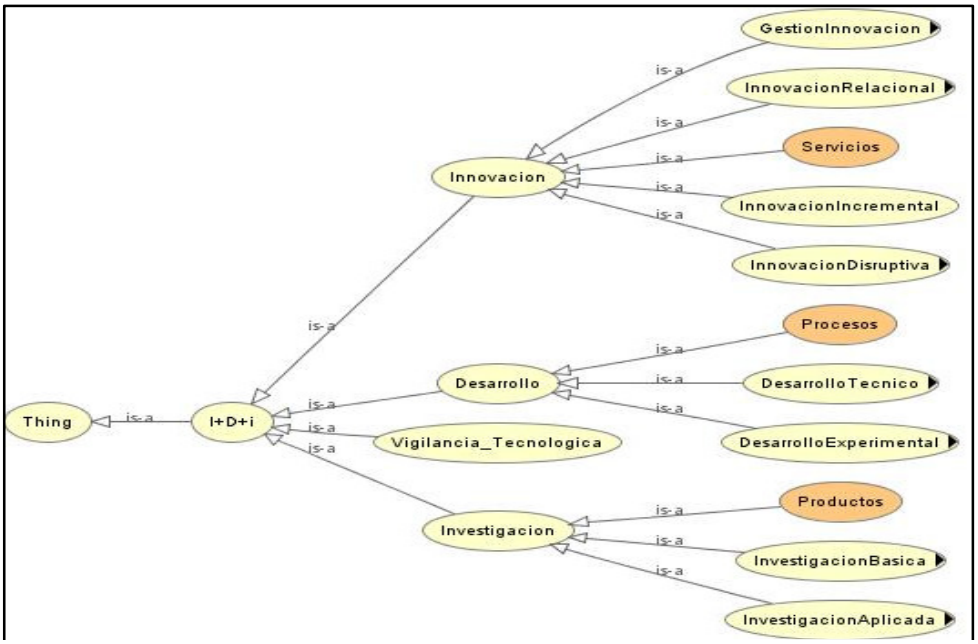


Figure 2. Partial View of Ontology R&D

Organizational learning allows for understanding the impact that perceptions and opinions; shared by R+D group human resources, have in a series of experiences or pieces of knowledge, for instance, technological surveillance. An R+D group can perform regular offline analysis, writing reports using Organizational Memory data analysis (OM), retrieved and formalized in real time. This model makes easier to incorporate the great volume of spontaneous and real time information provided by social networks, forums and blogs to assess its impact on trends and thematic behavior, so that both critical events and competitive advantages could be discovered.

Each learning level receives all content packages at intervals (e.g. daily, weekly) and analyzes them to determine what is mentioned in the R+D group social networks related to social and technological variables (e.g. positive, negative, neutral feelings) about issues such as technological surveillance.

The correlational analysis, once finished, is combined with statistical methods, such as factor analysis, both of which could be used to get merged trends in each lesson learned. Lessons learned are regarded as named entities, defined categories and, relevant and irrelevant topics in an R+D group; those results might be eventually used to calculate aggregates, identify trends and, to write reports, dashboards, and performance measurements.

On the other hand, social networks such as Facebook, LinkedIn and Twitter become a potential source of information. Their generalized use is spread worldwide, and they generate a huge amount of information related to R+D groups, which is useful as a supply for the knowledge

management model. The structure and organization of lessons learned, regarded as "discoveries" in the model, represent the relations between the results of a process, a project, an indicator, a condition or a cause which have eased and/or blocked the R+D groups strategic planning. Generally, it is recommended to describe such a discovery as a past event, even though it could be represented in the present whenever its effects and/or contexts keep being valid.

The knowledge management model continues with the implementation of a *syntactical / morphological* process of analysis, using the ontology-lexicon variation method, proposed by Mari (2009), which has been combined with NTKL (Natural Tool Kit Language), a natural language processing tool. When using NLTK, it was possible to separate each extracted lesson learned. The tree grammar decomposition described in Figure 3 tries to show the semantic behavior for each word, each one regarded as an entity contained in the R+D data ontology; articles, connectors and linking words have been discarded in this analysis, since they are not part of the terminology ensemble considered in the Ontology.

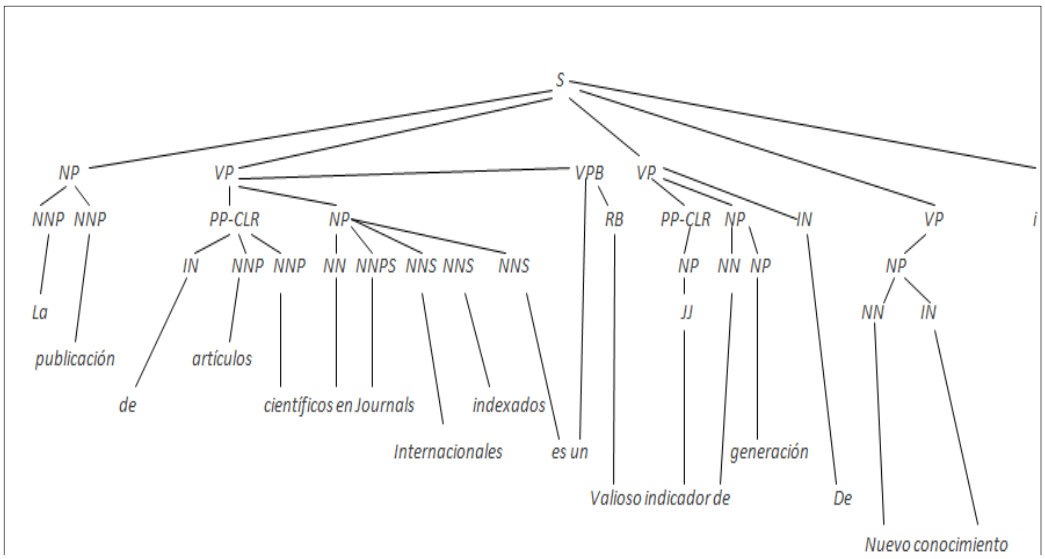


Figure 3. Lexical/morphological decomposition of a lesson learned taken from a social network

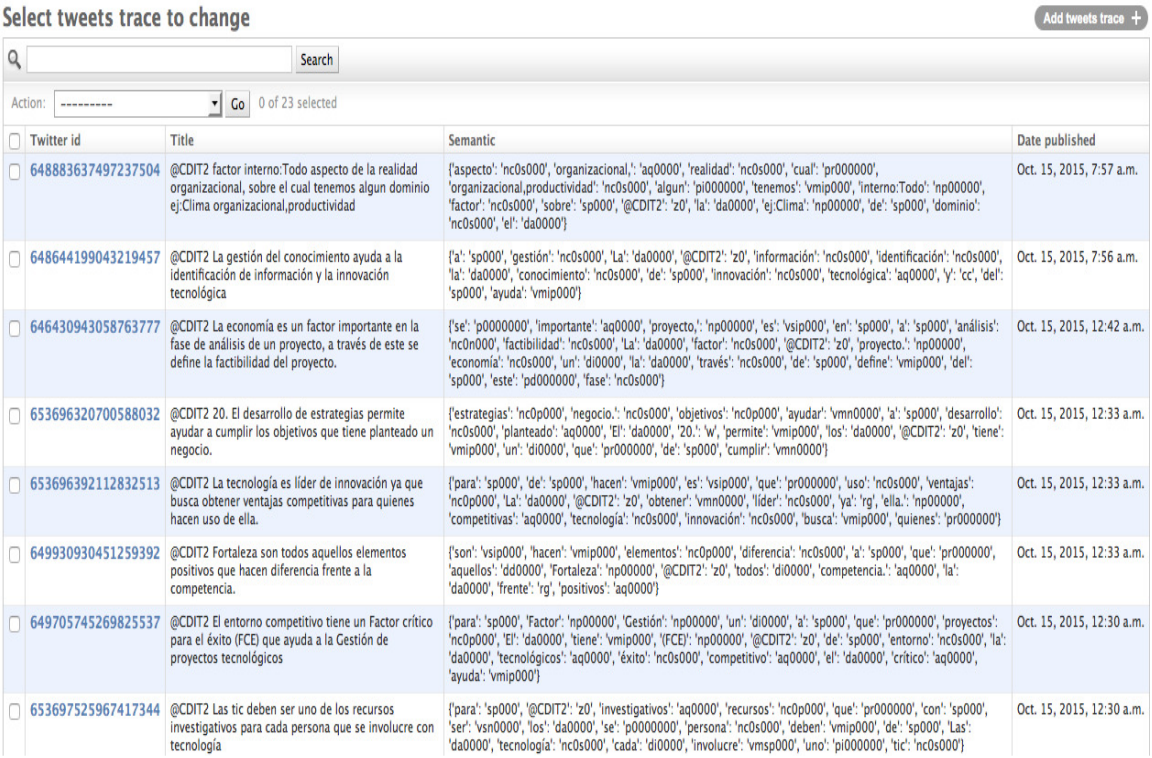
In the syntactic/morphologic process of lessons learned, Spanish language grammar rules are applied, and it considers punctuation marks as key syntactic items, so that the reading of each one of the characters contained in texts related to lessons learned could be started/stopped. If each character *(c)* in a phrase is considered as a chain, and spaces separate each character chain are followed additionally, by a period and a space, then that phrase is considered as complying with the suggested structure.

$$P(c1\ c2\ c3..cn) = \prod_{i=1}^{t} P(\ ci|\mathrm{c1}\ldots\mathrm{ci}-1) \qquad (1)$$

Eventually, the tokenization process is observed in Figure 3, applied to a lesson learned: *"La publicación de artículos científicos en Journals internacionales indexados es un valioso indicador de generación de nuevo conocimiento"* ["Scientific articles publication in international journals is a valuable sign of new knowledge generation]. Now, after applying the tokenization process, it is necessary to provide meaning to the phrase; this semantic analysis scenario in each

lesson learned will be referred to as semantic tagging or noting. Figure 4 shows the semantic analysis through eagles tags, applied to the notation process in others lessons learned.

The noting or tagging process provides semantic content to the phrase; and it is based on the structure given in the syntactic/morphologic analysis. In other words, the subcategorization of items such as subject, verb, and predicates is attained at this stage, so that verbal arguments and their corresponding syntactic/semantic functions are identified; for example, the term *"la producción científica"* [the scientific production] could be considered as an instance of a nominative syntagma (Subject), but also, it could be read as a Nominative Predicate, based on our lexical structure.



Twitter id	Title	Semantic	Date published
648883637497237504	@CDIT2 factor interno:Todo aspecto de la realidad organizacional, sobre el cual tenemos algun dominio ej:Clima organizacional,productividad	{'aspecto': 'nc0s000', 'organizacional,': 'aq0000', 'realidad': 'nc0s000', 'cual': 'pr000000', 'organizacional,productividad': 'nc0s000', 'algun': 'pi000000', 'tenemos': 'vmip000', 'interno:Todo': 'np00000', 'factor': 'nc0s000', 'sobre': 'sp000', '@CDIT2': 'z0', 'la': 'da0000', 'ej:Clima': 'np00000', 'de': 'sp000', 'dominio': 'nc0s000', 'el': 'da0000'}	Oct. 15, 2015, 7:57 a.m.
648644199043219457	@CDIT2 La gestión del conocimiento ayuda a la identificación de información y la innovación tecnológica	{'a': 'sp000', 'gestión': 'nc0s000', 'La': 'da0000', '@CDIT2': 'z0', 'información': 'nc0s000', 'identificación': 'nc0s000', 'la': 'da0000', 'conocimiento': 'nc0s000', 'de': 'sp000', 'innovación': 'nc0s000', 'tecnológica': 'aq0000', 'y': 'cc', 'del': 'sp000', 'ayuda': 'vmip000'}	Oct. 15, 2015, 7:56 a.m.
646430943058763777	@CDIT2 La economía es un factor importante en la fase de análisis de un proyecto, a través de este se define la factibilidad del proyecto.	{'se': 'p0000000', 'importante': 'aq0000', 'proyecto,': 'np00000', 'es': 'vsip000', 'en': 'sp000', 'a': 'sp000', 'análisis': 'nc0n000', 'factibilidad': 'nc0s000', 'La': 'da0000', 'factor': 'nc0s000', '@CDIT2': 'z0', 'proyecto.': 'np00000', 'economía': 'nc0s000', 'un': 'di0000', 'la': 'da0000', 'través': 'nc0s000', 'de': 'sp000', 'define': 'vmip000', 'del': 'sp000', 'este': 'pd000000', 'fase': 'nc0s000'}	Oct. 15, 2015, 12:42 a.m.
653696320700588032	@CDIT2 20. El desarrollo de estrategias permite ayudar a cumplir los objetivos que tiene planteado un negocio.	{'estrategias': 'nc0p000', 'negocio.': 'nc0s000', 'objetivos': 'nc0p000', 'ayudar': 'vmn0000', 'a': 'sp000', 'desarrollo': 'nc0s000', 'planteado': 'aq0000', 'El': 'da0000', '20.': 'w', 'permite': 'vmip000', 'los': 'da0000', '@CDIT2': 'z0', 'tiene': 'vmip000', 'un': 'di0000', 'que': 'pr000000', 'de': 'sp000', 'cumplir': 'vmn0000'}	Oct. 15, 2015, 12:33 a.m.
653696392112832513	@CDIT2 La tecnología es líder de innovación ya que busca obtener ventajas competitivas para quienes hacen uso de ella.	{'para': 'sp000', 'de': 'sp000', 'hacen': 'vmip000', 'es': 'vsip000', 'que': 'pr000000', 'uso': 'nc0s000', 'ventajas': 'nc0p000', 'La': 'da0000', '@CDIT2': 'z0', 'obtener': 'vmn0000', 'líder': 'nc0s000', 'ya': 'rg', 'ella.': 'np00000', 'competitivas': 'aq0000', 'tecnología': 'nc0s000', 'innovación': 'nc0s000', 'busca': 'vmip000', 'quienes': 'pr000000'}	Oct. 15, 2015, 12:33 a.m.
649930930451259392	@CDIT2 Fortaleza son todos aquellos elementos positivos que hacen diferencia frente a la competencia.	{'son': 'vsip000', 'hacen': 'vmip000', 'elementos': 'nc0p000', 'diferencia': 'nc0s000', 'a': 'sp000', 'que': 'pr000000', 'aquellos': 'dd0000', 'Fortaleza': 'np00000', '@CDIT2': 'z0', 'todos': 'di0000', 'competencia.': 'aq0000', 'la': 'da0000', 'frente': 'rg', 'positivos': 'aq0000'}	Oct. 15, 2015, 12:33 a.m.
649705745269825537	@CDIT2 El entorno competitivo tiene un Factor crítico para el éxito (FCE) que ayuda a la Gestión de proyectos tecnológicos	{'para': 'sp000', 'Factor': 'np00000', 'Gestión': 'np00000', 'un': 'di0000', 'a': 'sp000', 'que': 'pr000000', 'proyectos': 'nc0p000', 'El': 'da0000', 'tiene': 'vmip000', '(FCE)': 'np00000', '@CDIT2': 'z0', 'de': 'sp000', 'entorno': 'nc0s000', 'la': 'da0000', 'tecnológicos': 'aq0000', 'éxito': 'nc0s000', 'competitivo': 'aq0000', 'el': 'da0000', 'crítico': 'aq0000', 'ayuda': 'vmip000'}	Oct. 15, 2015, 12:30 a.m.
653697525967417344	@CDIT2 Las tic deben ser uno de los recursos investigativos para cada persona que se involucre con tecnología	{'para': 'sp000', '@CDIT2': 'z0', 'investigativos': 'aq0000', 'recursos': 'nc0p000', 'que': 'pr000000', 'con': 'sp000', 'ser': 'vsn0000', 'los': 'da0000', 'se': 'p0000000', 'persona': 'nc0s000', 'deben': 'vmip000', 'de': 'sp000', 'Las': 'da0000', 'tecnología': 'nc0s000', 'cada': 'di0000', 'involucre': 'vmsp000', 'uno': 'pi000000', 'tic': 'nc0s000'}	Oct. 15, 2015, 12:30 a.m.

Figure 4. Lesson Learned Tokenization and Eagle Tagging application

We use SPARQL for recovery and querying of these scenarios in organizational memory; given that data structure is organized by RDF syntax. SPARQL queries apply the method of a priori association rules specified by (Lin, He, & Everson, 2011); the association rules are given by the relation subject (S), verb (V), object (O) according to:

$$(S, V, O) \qquad (2)$$

The querying and interrogation to organizational memory aims to identify trends, issues, and feelings among others topics in corporate social networks that refer to themes of R & D. The process relies on time ranges intervals where desired interrogate organizational memory. An example of this can be: from September 1, 2015 to September 3, 2015 (01/09 / 2015□03 / 09/2015). In addition, this research integrates the relationship of four axis of consulting the

organizational memory; these axis are common strategic priorities in each of the development plans R & D research centers analyzed; each axis correspond to the following strategic areas: scientific and technological areas, indicators R & D (KPI), management projects and final results research R&D. Additionally, in the process of querying the organizational memory it arises retrieve information about:

- Actions: events associated with entities and people, represented on verbs
- Entities: the basic text units; i.e. places,
- Subject: represents people, objects, proper names and other lexical features that could be grouped under the subject, etc.
- Concepts: entities concepts, this element might be contained in predicates.
- Relations established among entities.
- Events where entities are involved.
- Feelings.

Table 1 shows two query scenarios for two lessons learned semantically different. The first scenario is related to the strategic axis *"scientific and technological areas";* the results shows that the semantic querying on the subject related ICT has relationship with scientific and technological area of the lesson learned; also it determined that the category to which reference is made corresponds to knowledge.

The other hand, the next scenario the lesson learned is related with research results R&D; the querying want determine which are the subject (resources), verb(action) and object(description). The result shows that semantically these elements correspond to "patente", "realizar", and "invento".

Table 1. Scenarios for querying and retrieval information from Organizational Memory

Scenario	Lesson learned	SPARQL Querying	Result
1	*Las TICs en la actualidad son fuente de conocimiento y de sabiduría de muchas personas que han generado y compartido sus logros*	**SELECT ?sujeto ?recurso WHERE { ? area rdfs:subClassOf ? ciencia }**	**tic | conocimiento**
Scenario	**Lesson learned**	**SPARQL Querying**	**Result**
2	*Nuestra patente es un contrato entre la Sociedad y el inventor individual, se realiza para que no allá ningún cambio del invento.*	**SELECT ?sujeto ?verbo ? objeto WHERE { ? categoria indicador i+d+i rdfs:subClassOf ? eje estrategico }**	**patente | realizar | invento**

5. CONCLUSIONS AND FUTURE WORK

Reviewing the literature reveals that the lessons learned, in any type of project, allow a team or working group to investigate how their dynamic capabilities for managing knowledge are being generated, and what the profile being developed is. This is done via a systematic analysis of individual or personal profiles of each of its members on a timeline. As some authors referenced in this paper conclude, future studies should focus on developing adequate and comprehensive work environments, where each person or individual can share knowledge and be provided with easier knowledge flows. This would allow for the transformation of tacit knowledge into explicit knowledge, and also for working with learning and development objectives in the contexts or fields being developed at an R&D center.

As many of the referred authors have concluded, the next step is providing proper and comprehensive spaces where every person and individual could share their knowledge, and, facilitating the flow of that knowledge. This will enhance the capacity of transforming tacit knowledge into explicit knowledge, which will allow for collaboration on learning objectives and development in contexts and areas where knowledge is generated in a research group.

In the same way, an organizational learning model design and validation is suggested, which will take into account lessons learned as a source of information, specifically, those circulating in corporate social networks related to R+D groups. The establishment of a uniform text structure is foreseen for all lessons learned about issues such as technological surveillance. Additionally, it is essential to organize a data ontology that allows for a semantic analysis of each lesson learned to build a knowledge database at an organizational memory level. This model will become a useful support tool for research groups for strategic planning and decision taking. It is expected a closer work with research groups in accredited universities in Bogotá D.C., in order to validate these results. The quantity of people in charge of technology surveillance and management will be determined for each research group.

Finally, lessons learned organization should respond to a readable textual format in a natural language process where an analysis could be easily performed. The metamodel will use a logical sequence of data, information and knowledge for a real time analysis on each one of these three elements. These elements circulate across all entities in the model. A representative sample of institutional documents, such as forms and formats is taken, where they are considered important for knowledge capturing.

ACKNOWLEDGEMENTS

The author would like to thank to Colombian Fund for Development of Higher Education (Fondo para el Desarrollo de la Educacíon Superior-FODESEP)

REFERENCES

[1] Tanner, D, (2014), “Creativity and Innovation in R&D”, R&D Innovator, pp.101-150.

[2] OCyT, ocyt.org. Retrieved February 24, 2015, from http://ocyt.org.co/es-es/

[3] MinTics. Mintics.gov.co, Retrieved February 26 , 2015, from http:// Mintics.gov.co/

[4] Bisadi, M, (2012) "Future Research Centers: The place of creativity and innovation", Procedia Social and Behaviors Sciences, Vol. 12, pp. 232-243.

[5] Hamid, T. & and Mehdi Jabbari Mohammed, (2011) "Product Innovation Performance in Organization", Procedia Technology, Vol. 1, pp. 521 – 523.

[6] Wiing, K. (2007) "Integrating Intellectual Capital and Knowledge Management". Long Range Planning, pp. 399-405.

[7] Pirró, G., Mastroianni, C., & Talia, D. (2010) "A framework for distributed knowledge management: Design and implementation". Future Generation Computer Systems, pp.38-49.

[8] Richter, P. & R. Weber, (2013), Case-Based Reasoning, Springer, pp.53-84.

[9] Michael, S & Davidsen, P. (2012)"How can organizational learning be modelled and measured?", Evaluation and Program Planning, Vol. 10, pp. 63-69.

[10] Bermell-Garcia P. Verhagen, W. & Astwood, S., (2012) "A framework for management of Knowledge Based Engineering applications as software services: Enabling personalization and codification". Advanced Engineering Informatics, Vol 20, pp. 219-230.

[11] Shehzad Rizwan & Naeem Muhammed, (2013) "Integrating knowledge management with business intelligence processes for enhanced organizational learning", International Journal on Software Eng, Vol 5, pp. 83-92.

[12] Caeir, M., Llamas, M., & Anido, L. (2014) "Computer Standards & Interfaces". Computer Standards & Interfaces, pp. 380–396.

[13] Greve, H. (2013), Microfoundations of Management: Behavioral Strategies and Levels of Rationality in Organizational Action. Academy of Management Perspectives, vol. 12, pp.103-119.

[14] Martinet, M, (2013) "L'intelligence economique. Les yeux et les oreilles de l'entreprise", París: Editions d'Organisation, Vol. 1, pp. 123-127.

[15] Corcho, O. (2010) "Construcción de ontologías legales con la metodología METHONTOLOGY y la herramienta WebODE". Law and the Semantic Web. Legal Ontologies, Methodologies, Legal, Information Retrieval, and Applications, Vol. 2, pp. 142-157.

[16] Ammann Eckhard, Ruiz-Montiel Marcela & Navas-Delgado Ismael, (2010) "Knowledge Development Conception and its Implementation: Knowledge Ontology", Rule System, COGNITIVE, Vol 2, pp. 60-65.

[17] Makkonen Teemu & Inkinen Tommi, (2013) "Innovation quality in knowledge cities: Empirical evidence of innovation award competitions in Finland", Expert Systems with Applications, Vol 2 pp. 5597–5604.

[18] Gomez-Perez, J.M & Corcho, O. (2008) "Problem-Solving Methods for Understanding Process Executions", Computing in Science & Engineering, Vol. 10, no. 3, pp. 47-52.

[19] King, W. (2009) "Knowledge Management and Organizational Learning", Annals of Information System, Vol. 4, pp. 3-13.

AUTHOR

Marco Javier Suárez Barón

He is Research at Unitec University from Bogota Colombia. He was borned in Duitama-Colombia; your works involves subject about machine learning, semantic web and knowledge discovery. Additionally the author received PhD in Strategic Planning and technology management at UPAEP México.

AUTHOR INDEX