FUTURE COMPUTING 2014

The Sixth International Conference on Future Computational Technologies and Applications

ISBN: 978-1-61208-339-1

May 25 - 29, 2014

Venice, Italy

FUTURE COMPUTING 2014 Editors

Kendall E. Nygard, North Dakota State University - Fargo, USA

Dan Tamir, Texas State University, USA

FUTURE COMPUTING 2014

Foreword

The Sixth International Conference on Future Computational Technologies and Applications (FUTURE COMPUTING 2014), held between May 25-29, 2014 in Venice, Italy, targeted advanced computational paradigms and their applications. The target was to cover (i) the advanced research on computational techniques that apply the newest human-like decisions, and (ii) applications on various domains. The new development led to special computational facets on mechanism-oriented computing, large-scale computing and technology-oriented computing. They are largely expected to play an important role in cloud systems, on-demand services, autonomic systems, and pervasive applications and services.

We take here the opportunity to warmly thank all the members of the FUTURE COMPUTING 2014 Technical Program Committee, as well as all of the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to FUTURE COMPUTING 2014. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the FUTURE COMPUTING 2014 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that FUTURE COMPUTING 2014 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the area of future computational technologies and applications.

We are convinced that the participants found the event useful and communications very open. We hope that Venice, Italy, provided a pleasant environment during the conference and everyone saved some time to enjoy the charm of the city.

FUTURE COMPUTING 2014 Chairs:

Cristina Seceleanu, Mälardalen University, Sweden
Hiroyuki Sato, The University of Tokyo, Japan
Miriam A. M. Capretz, The University of Western Ontario - London, Canada
Kendall E. Nygard, North Dakota State University - Fargo, USA
Vladimir Stantchev, SRH University Berlin - Institute of Information Systems, Germany
Wail Mardini, Jordan University of Science and Technology, Jordan
Alexander Gegov, University of Portsmouth, UK
Francesc Guim, Intel Corporation, Spain
Wolfgang Gentzsch, The UberCloud, Germany
Noboru Tanabe, Toshiba Corporation, Japan

FUTURE COMPUTING 2014

Committee

FUTURE COMPUTING Advisory Chairs

Cristina Seceleanu, Mälardalen University, Sweden
Hiroyuki Sato, The University of Tokyo, Japan
Miriam A. M. Capretz, The University of Western Ontario - London, Canada
Kendall E. Nygard, North Dakota State University - Fargo, USA
Vladimir Stantchev, SRH University Berlin - Institute of Information Systems, Germany
Wail Mardini, Jordan University of Science and Technology, Jordan
Alexander Gegov, University of Portsmouth, UK

FUTURE COMPUTING 2014 Industry/Research

Francesc Guim, Intel Corporation, Spain
Wolfgang Gentzsch, The UberCloud, Germany
Noboru Tanabe, Toshiba Corporation, Japan

FUTURE COMPUTING 2014 Technical Program Committee

Francisco Arcas Túnez, Universidad Católica San Antonio, Spain
Ofer Arieli, The Academic College of Tel-Aviv, Israel
Cristina Alcaraz, University of Malaga, Spain
Mohsen Askari, University of Technology Sydney, Australia
Abdelhani Boukrouche, University of Guelma, Algeria
Radu Calinescu, Aston University-Birmingham, UK
Alberto Cano, University of Córdoba, Spain
Miriam A. M. Capretz, The University of Western Ontario - London, Canada
Massimiliano Caramia, University of Rome "Tor Vergata", Italy
Jose M. Cecilia, Universidad Católica San Antonio, Spain
Chin-Chen Chang, Feng Chia University, Taiwan, R.O.C.
Cheng-Yuan Chang, National United University, Taiwan
Sung-Bae Cho, Yonsei University, Korea
Rosanna Costaguta, Universidad Nacional de Santiago del Estero, Argentina
Zhihua Cui, Taiyuan University of Science and Technology - Shanxi, China
Marc Daumas, INS2I & INSIS (CNRS), France
Isabel Maria de Sousa de Jesus, ISEP-Institute of Engineering of Porto, Portugal
Leandro Dias da Silva, Federal University of Alagoas, Brazil
Qin Ding, East Carolina University, USA
Prabu Dorairaj, NetApp Inc, India
Marek J. Druzdzel, University of Pittsburgh, USA
Fausto Fasano, University of Molise, Italy
Dietmar Fey, Friedrich-Alexander-University Erlangen-Nuremberg, Germany
Francesco Fontanella, Università degli Studi di Cassino e del Lazio Meridionale, Italy
Félix J. García Clemente, University of Murcia, Spain

Copyright Information

Table of Contents

A Novel Realization of Sequential Reversible Building Blocks

Vishal Pareek, Shubham Gupta
Dept. of Computer Science & Engineering
University College of Engineering
Kota, India
e-mail: {vishalpareekrk, guptashubham396}@gmail.com

Sushil Chandra Jain
Dept. of Computer Science & Engineering
Rajasthan Technical University
Kota, India
e-mail: scjain1@yahoo.com

Abstract—**With phenomenal growth of high speed and complex computing applications, the design of low power and high speed logic circuits have created tremendous interest. Reversible computing has emerged as a solution for future computing. A number of combinational circuits have been developed but the growth of sequential circuits was not significant due to feedback and fan-out was not allowed. However allowing feedback in space, sequential logic blocks have been reported in literature. The target technology is likely on quantum computing devices. Reversible flip-flops are the most significant and basic memory elements that will be the target building block of memory for the forthcoming quantum computing devices. This paper proposes a novel reversible gate and its quantum realization. The design of reversible flip-flops, Serial In Parallel Out (SIPO) shift register and shift counter is shown by using our proposed gate and basic reversible gates. The proposed design of sequential reversible circuits has significant improvement over earlier designs in terms of quantum cost and hardware complexity. It is expected that it will enhance the growth of sequential reversible circuit. The proposed gate is also parity preserving gate. This characteristic of the gate may also be useful in fault tolerant sequential circuit design.**

Keywords-reversible computing; sequential circuit; flip-flops; quantum computation.

I. INTRODUCTION

Heat dissipation and high power consumption are one of the most important issues in the digital circuit design. Conventionally, the logic elements are irreversible in nature. According to R. Landauer's principle [1], irreversible logic computation results in energy dissipation due to heat loss. Each bit of information dissipates at least KTln2 energy at which the operation is performed. In early 1973, C. H. Bennett [2] had shown that the problem of heat dissipation of VLSI (Very Large Scale Integrated) circuits can be overcome by using reversible logic. Due to this fact, the loss of information and consequently dissipation of energy in computational operation is significantly lower than conventional logic. Thus, reversible logic and its applications have spread in various technologies like low power CMOS (Complementary Metal-oxide Semiconductor) design, optical information processing, quantum computing, nanotechnology, etc.

In the design of reversible logic circuits, research was limited to the design of combinational circuits, due to the convention that feedback is not allowed in reversible computing. However, in one of the well known fundamental paper, T. Toffoli [3] has shown that feedback can be allowed in reversible computing. According to T. Toffoli," a sequential network is reversible if its combinational part (i.e., the combinational network obtained by deleting the delay elements and thus breaking the corresponding arcs) is reversible." In 1982, Edward Fredkin [4] has used this concept to propose the first design of the reversible sequential circuit called the JK latch having the feedback loop from the output to input. Recently, A. Banerjee et al. [5] have redefined that feedback is allowed in space but not in time. Hence, the development of reversible sequential circuits has begun.

In the current literature, the significant work can be found on the efficient design of combinational reversible circuits such as full adders, BCD (Binary Coded Decimal) adders, encoder and multiplexers and several synthesis methods have been proposed [6]. In last few years, the design of sequential reversible circuits has attracted the attention of researchers in the area of optimization and synthesis.

A Reversible Gate is a p-input, p-output (denoted by p*p) circuit that produces a unique output pattern for each possible input pattern. There is a one to one correspondence between the input and output vectors. Any reversible logic design should minimize the following optimization parameters:

a) Gate Counts: The total number of gates used in a circuit.

b) Garbage Outputs: These Outputs are not used in output functions, but are required to maintain the reversibility. The garbage outputs are defined for a circuit not for a gate.

ISBN: 978-1-61208-339-1

c) Constant Inputs: Constants are the input lines that are either set to zero or one in the circuit's input side.

d) Quantum Cost: Each reversible gate has a cost associated with it called the Quantum Cost. The quantum cost of a reversible gate is the number of 1×1 and 2×2 reversible gates or quantum logic gates required in its design. The computational complexity of a reversible gate can be represented by its quantum cost. The quantum costs of all reversible 1×1 and 2×2 gates are taken as unity.

e) Hardware Complexity: The total number of logic operations in a circuit is known as hardware complexity. In hardware complexity, the terms are:

α = A two input EX-OR gate calculation

β = A two input AND gate calculation

δ = A NOT calculation

Basically, it refers to the total number of AND, OR and EXOR operation in a circuit.

However, the gate count is not a good metric of optimization as reversible gates are of different type having different computational complexity. Hence, the optimization of quantum cost is the major metric in the design of sequential reversible circuits. A. Banerjee et al. [7] have addressed that if a set of reversible quantum gates organized in a black box then it can be visualized as a new gate. Reduction of these parameters should be the main design focus in sequential reversible circuits. The basic reversible gates as Toffoli Gate [3] (CCNOT gate), Feynman Gate (CNOT Gate) [8] and Fredkin Gate [4] will be used in the designing of sequential reversible circuits.

In this work, we are proposing a novel parity preserving reversible gate and its quantum realization by using quantum cost optimization algorithm [7]. Further, our propose gate is used in the realization of reversible flip-flops and shift counter.

The next sections of this paper are as follows. Section II provides the related work in the past. Section III provides the details about the proposed reversible gate and its quantum realization. Section IV provides reversible design of flip-flops. The reversible design of SIPO shift register and shift counter is described in section V. Section VI has the discussion on results. Section VII concludes the work.

II. RELATED WORK

In 2005, the first attempt on the design of reversible flip-flop was H. Thapiyal et al. [9]. In this work, the Fredkin, Feynman and New Gate was used as AND, NOT and NOR Gate respectively. In the designing of reversible flip-flop, the conventional design of a flip-flop was used. In 2006, J. E. Rice [10] has proposed all the reversible flip-flops (except R-S) using R-S latch. For the designing of reversible flip-flops, Toffoli and Feynman Gate were used as CCNOT and CNOT gate respectively. S. K. S. Hari et al. [11] have addressed reversible flip-flops by using basic reversible Fredkin and Feynman gates. The reversible flip-flops were proposed by A. Banerjee et al. [5] in 2007. For the construction of reversible flip-flops, Toffoli gate (CCNOT Gate), Feynman (CNOT gate) and NOT were used. A novel concept on the designing of reversible flip-flops was proposed by Min-Lun Chuang et al. [12] in 2008. In this work, all reversible latches (except the SR latch) and their corresponding flip-flops were proposed. In 2011, the reversible D flip-flop and shift registers [13] and T flip-flop were addressed by V. Rajmohan et al. [14]. Two Sayem Gates and one Fredkin Gate were used for the designing of reversible T flip-flop.

At last but not least, in 2012, reversible J-K and D flip-flop were proposed by Lafifa Jamal et al. [15] using basic reversible Fredkin and Double Feynman gates. Recently, the design of reversible T flip-flop was proposed by Shubham Gupta et al. [16] using a new gate named SVS gate.

Thus, from a careful survey of the existing works on reversible sequential circuits, it can be concluded that most of these work considered the optimization of number of reversible gates and garbage outputs, while ignoring the important parameters of quantum cost and hardware complexity.

Our goal is to describe the quantum realization of the proposed reversible gate and optimize the design of reversible sequential circuits in terms of all important parameters, viz., the gate counts, quantum cost, garbage outputs and hardware complexity. Normally, two or three parameters are optimized in earlier designs but we have considered all above parameters shown significant improvement in most of the cases using our proposed reversible gate. The low cost shift counter is also designed using the proposed reversible D flip-flop and basic reversible gates.

III. PROPOSED REVERSIBLE PARITY PRESERVING GATE AND ITS QUANTUM REALIZATION

The section describes our proposed parity preserve gate named "Pareek" gate and its quantum realization.

A. Proposed Reversible Parity Preserving Gate

We propose a new 4×4 parity preserve reversible circuit, called Pareek gate. The block diagram of the proposed gate is shown in Fig. 1.

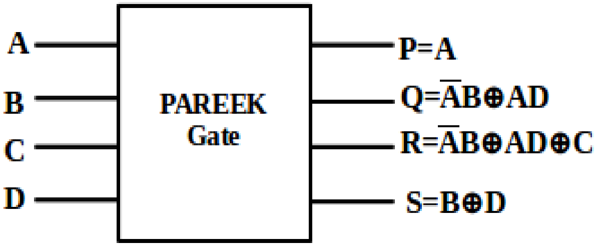


Figure 1. Proposed parity preserving reversible pareek gate

The truth table of proposed parity preserving Pareek Gate is shown in Table I.

The output P (=A) is copied directly from input A, this input to output line is called control line where as other lines are called target lines. The gate produces three outputs, namely, Q, R and S on target lines as defined in Fig. 1. The outputs are verified manually through the truth table.

TABLE 1. TRUTH TABLE OF THE PROPOSED REVERSIBLE GATE

Input				Output			
A	**B**	**C**	**D**	**P**	**Q**	**R**	**S**
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	1	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	0	0	0
1	0	0	1	1	1	1	1
1	0	1	0	1	0	1	0
1	0	1	1	1	1	0	1
1	1	0	0	1	0	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	0	1	1
1	1	1	1	1	1	0	0

It is observed that the parity of the input bits is equal to the parity of the output bits in each row of the Table I. Hence, the gate also preserves the parity. This characteristic can further be used in fault tolerant design of reversible sequential circuit. However, we propose the low cost design of reversible sequential building blocks using this gate.

B. Quantum Realization of Proposed Gate

The quantum cost of a reversible gate is the number of 1×1 and 2×2 reversible gates or quantum logic gates required in its design. The computational complexity of a reversible gate can be represented by its quantum cost. The quantum costs of all reversible 1×1 and 2×2 gates are taken as unity. Any reversible gate can be realized using the 1×1 NOT gate, and 2×2 reversible gates such as Controlled-V and Controlled-V$^+$ and the Feynman gate which is also known as the Controlled NOT gate (CNOT). Thus, it can said that the quantum cost of a reversible gate can be calculated by counting the numbers of NOT, Controlled-V, Controlled-V$^+$ and CNOT gates required in its implementation. The quantum cost of proposed reversible gate is calculated by an optimization algorithm [7].

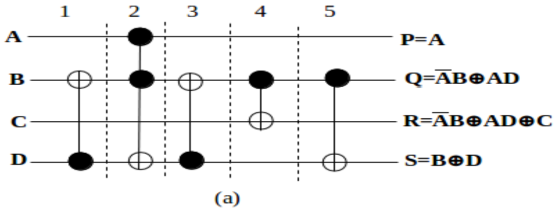

1
2
3
4
5
A
B
C
D
P=A
$Q=\overline{A}B\oplus AD$
$R=\overline{A}B\oplus AD\oplus C$
$S=B\oplus D$
(a)

Figure 2. (a) Realization of proposed gate using Toffoli and CNOT gate

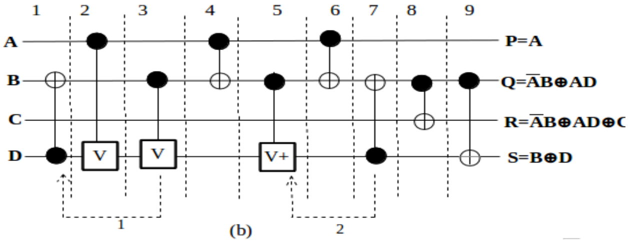

1
2
3
4
5
6
7
8
9
A
B
C
D
V
V
V+
1
2
(b)
P=A
$Q=\overline{A}B\oplus AD$
$R=\overline{A}B\oplus AD\oplus C$
$S=B\oplus D$

Figure 2. (b) Quantum realization of proposed gate

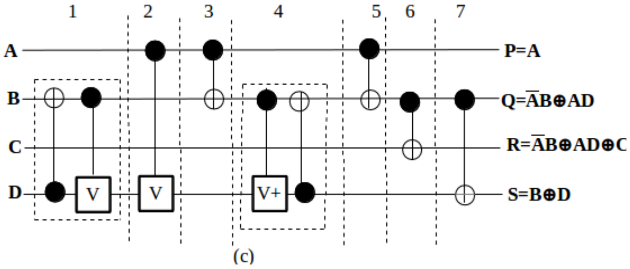

1
2
3
4
5
6
7
A
B
C
D
V
V
V+
(c)
P=A
$Q=\overline{A}B\oplus AD$
$R=\overline{A}B\oplus AD\oplus C$
$S=B\oplus D$

Figure 2. (c) Optimized quantum realization of proposed gate

In Fig.2 (a), the proposed Pareek gate is realized using one Toffoli gate and four CNOT gates. Then, Toffoli and CNOT gates are substituted by quantum primitives and moving rule is applied (the movements are shown by arrows), so its direct linear cost is (1×5) + (4×1) =9, which is shown in Fig.2 (b). New gates are introduced (dashed boxes) in Fig.2 (c) to yield quantum cost of Pareek gate as 7.

IV. PROPOSED REVERSIBLE FLIP-FLOPS

A flip-flop is a circuit that has two stable states and can be used to store state information. It is the basic storage element in sequence logic. The design of reversible flip-flops is proposed in this section.

A. Proposed Reversible D flip-flop

We propose the realization of D Flip-flop using our proposed reversible gate. The reversible design is shown in Fig. 3 and the corresponding block diagram is shown in Fig.4.

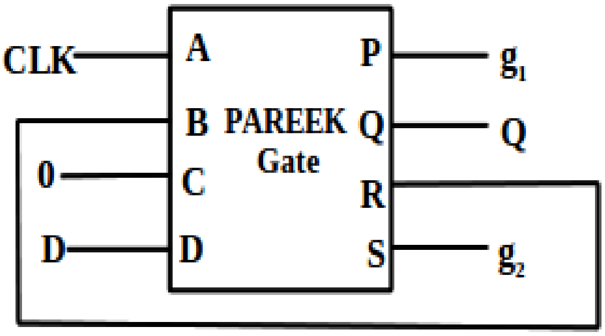


Figure 3. Proposed Realization of Reversible D flip-flop

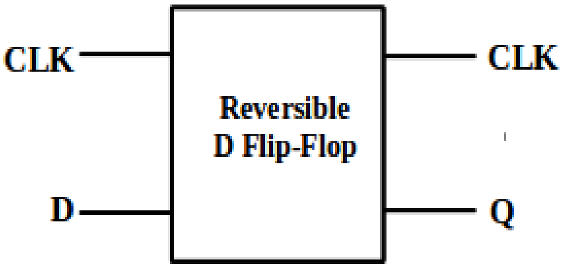


Figure 4. Proposed Block Diagram of Reversible D Flip-Flop

Due to the proposed parity preserving gate, the realization of reversible D flip-flop is also parity preserving.

B. Proposed Reversible R-S flip-flop

The reversible design of R-S flip-flop is shown in Fig. 5.

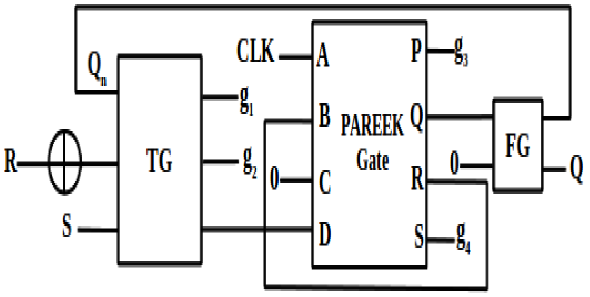


Figure 5. Proposed Realization of Reversible R-S Flip-Flop

The realization of R-S Flip-flop is proposed using our proposed reversible gate, Feynman and Toffoli Gate.

C. Proposed Reversible J-K flip-flop

A J-K flip-flop is a refinement of the R-S flip-flop in that the indeterminate state of the R-S type is defined in the J-K type. The reversible design is shown in Fig. 6.

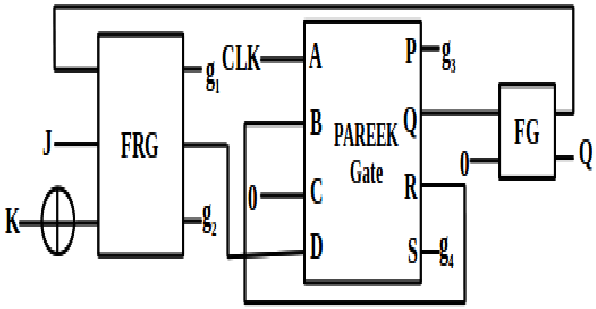


Figure 6. Proposed Realization of Reversible J-K Flip-Flop

The proposed reversible J-K Flip-flop is realized using our proposed reversible gate, Fredkin and Feynman Gate.

D. Proposed Reversible T flip-flop

The T flip-flop is a single-input version of the J-K flip-flop. The reversible design of proposed T flip-flop is shown in Fig.7.

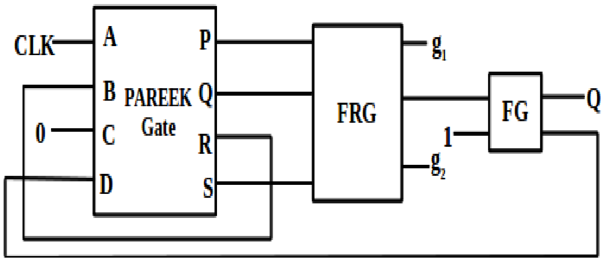


Figure 7. Proposed Realization of Reversible T Flip-Flop

T Flip-flop is realized using our proposed reversible gate, Fredkin and Feynman Gate.

V. DESIGN OF PROPOSED REVERSIBLE SIPO SHIFT REGISTER & SHIFT REGISTER COUNTER

The shift register is an indispensable functional device in a digital system. A register capable of shifting binary information either to the right or to the left is called shift register. In a shift register, the flip-flops are connected in such a way that the bits of a binary number are entered into the shift register, shifted from one position to another and finally shifted out.

Shift register can be arranged to form counters. Shift register counters use feedback, whereby the output of the last flip-flop in the shift register is connected back to the first flip-flop.

This section provides the reversible design of Serial In and Parallel Output (SIPO) shift register and shift register counter by using our proposed reversible gate.

A. Proposed Reversible SIPO Shift Register

A 4-bit Serial in parallel out shift register consists of one serial input, and outputs are taken from all the flip-flops parallel. In this register, data is shifted in serially but shifted out in parallel. The reversible design of SIPO shift register using proposed D flip-flop is shown in Fig. 8.

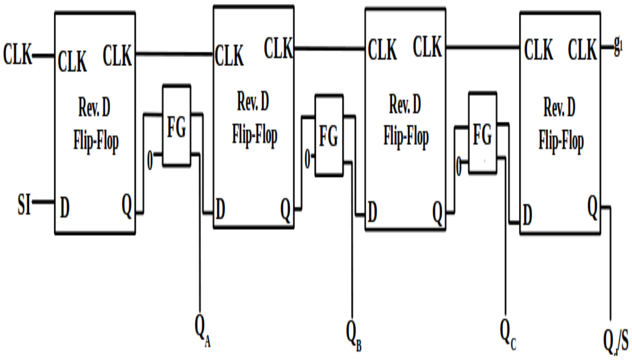


Figure 8. Proposed Realization of Reversible SIPO Shift Register

The serial input is provided to the SI input of the reversible left-most flip-flop while the outputs Q_A, Q_B, Q_C, Q_D are available in parallel from the Q output of the flip-flops.

B. Proposed Reversible Shift Counter (Johnson Counter)

In shift counter, the inverted true output (Q) of the last flip-flop is connected back to the serial input of the first flip-flop. Fig. 9 shows reversible design of shift counter using proposed reversible D flip-flop.

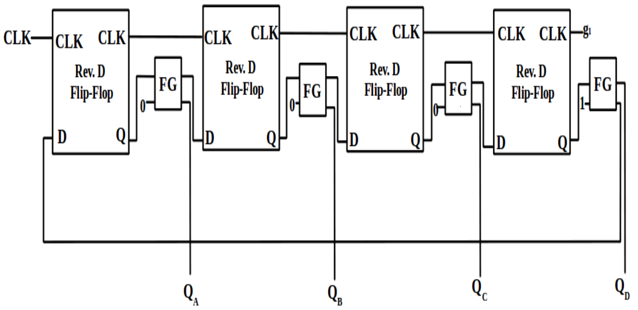


Figure 9. Proposed Realization of Reversible Shift Counter

The output of each reversible flip-flop (Q) is connected to the D input of the next stage. However, the inverted output of the last flip-flop.

VI. DISCUSSION ON RESULTS

Table 2 to Table 7 shows statistics and comparison of our new proposed design of sequential elements against the proposed designs by various researchers. We use the optimization parameters like gate counts, garbage output, constant input, quantum cost and hardware complexity as the cost functions to measure the quality of the design.

The row for "Improvement Percentage" (IP) is the percentage factor of {100-(Proposed Design/Existing Design)*100} %. For example, for a D flip-flop design, our realization has 1 gate while Existing Design [10] has 11 gates. Thus, the ratio is {100-(1/11)*100} % = 91%. The cost of constant input, quantum cost and hardware complexity is not summarized by all the researchers of their reversible sequential elements. We count these numbers based on their designs and show them in tables.

TABLE 2. STATISTICS & COMPARISON OF REVERSIBLE D FLIP-FLOP OVER VARIOUS OPTIMIZATION PARAMETERS

	Gate Count	Garbage Output	Constant Input	Quantum Cost	Hardware Complexity
Existing Design[10]	11	12	12	47	16α+24β+5δ
Existing Design[11]	4	3	2	12	6α+8β+2δ
Existing Design[12]	5	3	2	13	6α+8β+3δ
Existing Design[13]	1	2	1	8	4α+4β+δ
Proposed Design	***1***	***2***	***1***	***7***	***3α+2β+δ***
IP w.r.t.[10]	91%	83.3%	91.6%	85%	Improved
IP w.r.t.[11]	75%	33.3%	50%	41.6%	Improved
IP w.r.t.[12]	80%	33.3%	50%	46%	Improved
IP w.r.t.[13]	0%	0%	0%	12.5	Improved

TABLE 3. STATISTICS & COMPARISON OF REVERSIBLE R-S FLIP-FLOP OVER VARIOUS OPTIMIZATION PARAMETERS

	Gate Count	Garbage Output	Constant Input	Quantum Cost	Hardware Complexity
Existing Design[5]	9	6	5	33	8α+6β+δ
Existing Design[9]	6	8	6	18	10α+12β+8δ
Proposed Design	***4***	***4***	***2***	***13***	***5α+3β+δ***
IP w.r.t.[5]	55%	33%	60%	60%	Improved
IP w.r.t.[9]	33%	50%	66%	27%	Improved

TABLE 4. STATISTICS & COMPARISON OF REVERSIBLE J-K FLIP-FLOP OVER VARIOUS OPTIMIZATION PARAMETERS

	Gate Count	Garbage Output	Constant Input	Quantum Cost	Hardware Complexity
Existing Design[10]	12	14	13	52	17α+26β+5δ
Existing Design[11]	6	6	4	22	10α+16β+4δ
Existing Design[12]	7	4	3	27	7α+9β+2δ
Proposed Design	***4***	***4***	***2***	***13***	***6α+6β+3δ***
IP w.r.t.[10]	66%	71%	84%	75%	Improved
IP w.r.t.[11]	33%	33%	50%	41%	Improved
IP w.r.t.[12]	42%	0%	33%	51%	Improved

TABLE 5. STATISTICS & COMPARISON OF REVERSIBLE T FLIP-FLOP OVER VARIOUS OPTIMIZATION PARAMETERS

	Gate Count	Garbage Output	Constant Input	Quantum Cost	Hardware Complexity
Existing Design[10]	13	14	14	53	18α+26β+5δ
Existing Design[11]	5	3	2	13	7α+8β+2δ
Existing Design[14]	3	3	2	17	9α+8β+2δ
Proposed Design	***3***	***2***	***2***	***13***	***6α+6β+2δ***
IP w.r.t.[10]	76%	85%	85%	75%	Improved
IP w.r.t.[11]	40%	33%	0%	0%	Improved
IP w.r.t.[14]	0%	33%	0%	23%	Improved

TABLE 6. STATISTICS & COMPARISON OF REVERSIBLE SIPO SHIFT REGISTER OVER VARIOUS OPTIMIZATION PARAMETERS

	Gate Count	Garbage Output	Constant Input	Quantum Cost	Hardware Complexity
Existing Design[13]	7	5	7	35	19α+16β+4δ
Proposed Design	***7***	***5***	***7***	***31***	***15α+8β+4δ***
IP w.r.t.[13]	0%	0%	0%	11%	Improved

TABLE 7. STATISTICS OF REVERSIBLE SHIFT REGISTER COUNTER OVER VARIOUS OPTIMIZATION PARAMETERS

	Gate Count	Garbage Output	Constant Input	Quantum Cost	Hardware Complexity
Proposed Design	***8***	***5***	***8***	***32***	***16α+8β+4δ***

According to the statistics in Table 2 to Table 7, the implementation cost of our designs is lower than designs of all the existing designs in literature.

VII. CONCLUSIONS

In this paper, we have proposed a complete set of reversible sequential elements corresponding to available irreversible sequential designs. Our proposed reversible flip–flop and shift register realization are significantly improved over existing reversible realization in terms of gate count, garbage output, constant input, quantum cost and hardware complexity. We have also proposed low cost shift register counter design using our proposed gate. We have also proposed a quantum realization of our proposed gate. With this implementation, the power consumption of these reversible designs can be controlled and kept significantly low. Our proposed gate is parity preserving. This characteristic of the gate can also be used in fault tolerant sequential circuits designs which is still unexplored area of research.

REFERENCES

[1] R. Landauer, “Irreversibility and heat generation in the computing process,” IBM J. Research and Development, vol.5, no. 3, July,1961, pp. 183-191.

[2] C. H. Bennett, “Logical Reversibility of Computation,” IBM J. Research and Development, vol. 17, no. 6, November 1973, pp. 525-532.

[3] T. Toffoli,” Reversible computing,” Tech rep, MIT/LCS/TM-151, MIT Lab for Computer Science, 1980, pp. 632-644.

[4] E. Fredkin and T. Toffoli,”Conservative logic,” MIT/LCS/TS-545, Cambridge, Massachusetts, 1982, pp. 219–255.

[5] A. Banerjee and A. Pathak, “On the Synthesis of Sequential Reversible Circuit,” arXiv [quant-ph], 28 July 2007, pp. 1-9.

[6] D. Maslov and G.W. Dueck, “Reversible cascades with minimal garbage,” IEEE Trans.on Computer Aid. Des. of Integrated Circuits and Systems, vol. 23, no.11, November 2004, pp. 1497-1509.

[7] A. Banerjee and A. Pathak, “An Algorithm for Minimization of Quantum Cost,” Natural Sciences Publishing Corporation, An International Journal of Applied Mathematics & Information Sciences, 2012, pp. 157-165.

[8] R. Feynman, “Quantum Mechanical Computers,” Optical News, 1985, pp. 11-20.

[9] H. Thapiyal and M. B. Srinivas, “A Beginning in the Reversible Logic Synthesis of Sequential Circuits,” Proceedings of Military & Aerospace Programmable Logic Devices International Conference, 2005, pp. 1-5.

[10] J. E. Rice, “A New Look at Reversible Memory Elements,” Proceedings of IEEE International Symposium on Circuits and Systems, Lethbridge Univ., Alta, May 2006, pp. 1243-1246.

[11] S. K. S Hari, S. Shroff, S. N. Mahammad, and V. Kamakoti,”Efficient Building Blocks for Reversible Sequential Circuit design,” 49th IEEE International Midwest Symposium on Circuits and Systems,vol.1, 2006, pp. 437-441.

[12] M. L. Chuang and C. Y. Wang , “Synthesis of Reversible Sequential Elements,” ACM Journal on Emerging Technologies in Computing Systems, vol. 3, no. 4, Article 19, January 2008, pp. 1-19.

[13] V. Rajmohan and V. Ranganathan, “Optimized Shift Register Design Using Reversible logic,” 3rd International Conference on Electronics Computer Technology (ICECT), IEEE, vol.2, 2011, pp. 236-239.

[14] V. Rajmohan and V. Ranganathan, “Design of Counters Using Reversible Logic,” 3rd International Conference on Electronics Computer Technology (ICECT), IEEE, vol.5, 2011, pp. 138-142.

[15] L. Jamal, F. Sharmin, M. A. Mottalib, and H. M. H. Babu, “Design and Minimization of Reversible Circuits for a Data Acquisition and Storage System,” International Journal of Engineering and Technology, vol. 2, no. 1, January 2012, pp. 9-15.

[16] S. Gupta, V. Pareek, and S.C. Jain, “Low Cost Design of Sequential Reversible Counters,” International Journal of Scientific & Engineering Research, vol. 4, no. 11, November 2013, pp. 1234-1240.

Improved AODV Protocol to Detect and Avoid Black Hole Nodes in MANETs

Muneer Bani Yassein, Yaser Khamayseh, Bahaa Nawafleh
Department of Computer Science
Jordan University of Science and Technology
22110, Irbid, Jordan
Email: {masadeh, yaser}@just.edu.jo, nawafleh_bahaa@yahoo.com

***Abstract*— Security in Mobile Ad Hoc Networks (MANETs) is difficult to achieve because of the different attacks that might occur in the network, such as black hole attacks. In black hole attacks, the malicious node tries to attract most of the network traffic by advertising it has the best routing paths to the destination nodes, once the traffic is received by the black hole node, it simply drops the packets. This paper proposes an enhancement to Ad hoc On-Demand Distance Vector (AODV) routing protocol by employing effective policies to detect and avoid black hole nodes. The performance of the proposed scheme is evaluated using simulation. The obtained performance results indicate that the proposed AODV protocol achieves a significant improvement over both MI-AODV and the original AODV protocols, in terms of packet delivery ratio, dropped packets ratio, and overhead.**

***Keywords*- Black Hole; Routing Protocol; Mobile Ad hoc Networks; Wireless Network; AODV.**

I. INTRODUCTION

A wireless ad hoc network is a network using different airwaves (such as radio waves) to connect a collection of infrastructureless nodes; it differs from wired network which uses physical connection. Due to the open nature of wireless links, wireless links face many challenges, such as security, routing, and scheduling [3][6][12][14]. A Mobile Ad Hoc Network (MANET) is a group of mobile devices that are connected through wireless links. These nodes collaborate together in order to achieve different network functionalities. Moreover, it uses a point to point transmission and each node works as a host and as a router [1][3]. Each node in the network may be sender, receiver, or intermediate node that provides contact of the other nodes, and these networks do not have any infrastructure such as Base Stations. MANETs are vulnerable to attacks and threats [4][9]. The process of transmitting data between the source and the destination nodes through the path is called routing. The determination of the best path depends on many measurements such as paths cost, and number of hops.

Routing involves two sub processes, namely, (i), determining the best routing paths from the source to the destination, and (ii), transferring the data packets using the discovered path. Security in MANETs is difficult to achieve because of different attacks that might occur in the network (e.g., black hole attacks). In black hole attacks, the malicious node tries to attract as much as possible of the network traffic by advertising the best routing paths to the destination nodes, once the traffic is received by the black hole node, it drops the data. For example, in the widely used Ad hoc On-Demand Distance Vector (AODV) [4] routing protocol in a network infected with black hole nodes, when a source node sends data packets into a destination, the black hole advertises that it has the best path to the destination node whenever it receives any Route Request (RREQ) control packet. Then, it sends the response, Route Reply (RREP) to the source node. RREP messages could arrive from a normal node or a black hole node. If the reply arrives from a normal node, the protocol works as intended. If the first replay arrives from a black hole node, the source will transmit the data through the path that contains the black hole node. Once the data is received by a black hole node, it drops the data.

The probability of black hole node replies first to the RREQ message increases if the black hole is physically closer to the destination. Moreover, the probability of a false RREP message from a black hole arriving first to the source is higher than a normal safe reply as a black hole nodes response immediately to RREQ messages without the need for waiting a responding from the destination or checking the Routing Table (RT) as in the case of normal nodes.

According to the AODV specification, once a source receives a RREP message, this makes the routing discovery process completed, and thus, it ignores all other reply messages from other nodes, and it begins sending the data packets using the received path. This work aims at improving the AODV protocol in order to detect and avoid black hole nodes to improve data packet delivery ratio, to provide secure routing, and to increase the network performance.

The rest of this paper is organized as the following: Section II presents some of the related work in the area. Section III presents the proposed scheme. Section IV presents the simulation environment and the obtained results. The paper is concluded in Section V.

II. RELATED WORK

Different mechanisms were proposed to solve the black hole node problem. Most of researches conducted in this area can be divided into three categories: securing existing protocols, developing new secure protocols, and intrusion detection techniques. The following is a sneak review of some of the works that attempt to solve the black hole problem [8][16][17][18][20][21][22][23][25][24].

Sangi et al. [20] analyzed the performance degradation for AODV protocol, especially if the byzantine attacks are generated in a combination. In their analysis, they used

GloMoSim simulator. The authors concluded that the effects of byzantine and black hole are devastating when they are compared to a single black hole attack. In the routing protocol, route rushing or wormhole attacker maximizes the probability of malicious nodes. Also, a limited number of malicious nodes may generate wormhole attack with a combination between black/gray hole attack in which it may affect the activities of the network more than the rushing with black/gray hole attack.

Medadian et al. [21] proposed a new scheme to prevent the black hole attacks based on the discussion between neighbor nodes in the network that will participate in the communication between the source and the destination nodes. The proposed scheme provides a higher security and better performance in delivering packets than the traditional AODV. The proposed scheme restricts each node with a number of rules to identify if they are not attacker; the node activities within the network determine if it is honest or not, in order to be a participant in the transmission process, the node must proves its honesty. Firstly, every node in the network is allowed to be a participant of the transmission process between the source and destination nodes, so each node has enough time to prove its truth. Min and Jiliu [22] addressed the security issues included in the routing process in MANET networks, in addition to detecting multiple black holes that act in groups in the networks, it proposed two authentication approaches using hash functions: firstly, the Message Authentication Code (MAC), and secondly, the Pseudo Random Function (PRF). Based on these two approaches, it can be fast to verify the message and to identify the group, making it possible to determine multiple black holes that work and cooperate together, also to find the safe routing path while avoiding attack from black hole groups.

Zhang et al. [23] proposed a new approach for detecting black holes based on the process of checking a sequence number assigned to the Route Reply message based on the use of a new message generated by the destination of the route.

The proposed scheme is used to deal with malicious attacks and with the problems resulted from traditional methods, rather than using a public key as in the traditional methods in which this may result in extra problems, such as key distribution, instead, in this scheme, an intermediate node in the network unicast a message along with a defined control message to the destination to ask for up to date serial number. Khamayseh et al. [8] proposed the protocol MI-AODV to detect black hole nodes in a network. This mechanism modifies the original AODV protocol to enable the nodes of detecting black hole nodes in the network.

III. Proposed Protocol

The security issue in MANETs is essential and even more challenging because of multiple senders, multiple receivers, and the usage of wireless links for transmitting the data. Thus, MANETs are more likely to be affected by attacks, such as the black hole attack. In general, MANET attacks can be classified into two types; external (outside) and internal (inside) attacks. The external attacks are caused by nodes that do not belong to the domain of the network, while the internal attacks are caused by the nodes which are part of the network itself.

Furthermore, a black hole attack is a type of denial of service attack where a malicious node can attract data packets by falsely advertise a fresh route to the destination and retain them without forwarding them to the destination. This work proposes a mechanism for preventing the black hole attack by modifying the operations of the AODV routing protocol. The proposed mechanism aims at detecting and avoiding black hole nodes in MANET to reduce its impact. The proposed mechanism utilizes the following observations:

- The necessity to monitor the RREP messages and to observe its history. In this work, we propose to insert a new field in the RREP message to store the address of the last node that has a path to the destination.
- The necessity to observe the behavior of other nodes. Create new two tables in each node: suspect and black list tables.
- Suspect table contains the addresses of intermediate nodes which have sent RREP message; it also includes the number of times a node failed to send data through this node. For each node *i*, the suspect table contains a list of all nodes in the network that node *i* have received a RREP message and for node *i* the number of failures. A RREP message is considered failed if it was not able to deliver the data to the destination using the specified path. If the node doses not receive an acknowledgment message it considered the data is lost and restart the routing process again to retransmit the data.
- Black list table contains a list of nodes with failed RREP message that exceeded a certain threshold.

If node *i* receives a RREP message from node j, with invalid path, it adds node *j* to the suspect table. Once the number of failures for a particular node exceeds a certain threshold, this node is moved to the black list table and any coming RREP messages from this node will be ignored.

- Add acknowledgment message of length one bit. The message is set to 1 if the packets are delivered to the destination node; otherwise, it is set to 0. The acknowledgment message will be forwarded to the source node to acknowledge the recipient of the send data.

Moreover, the source sends a RREQ message in a standard manner as in the original ADOV protocol. In this scenario, the source node *S* sends RREQ to the destination *D* through intermediate nodes. When a RREP message is received from intermediate nodes, the following steps are performed:

- Transmit the data packets through the path received in the first route replay message. In Figure 1, the first RREP message arrives to the source node through intermediate nodes I3, I5. Figure 1 shows the first RREP arrived to the source node.

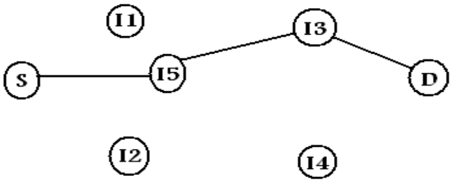


Figure 1. First RREP arrived to the source node.

- The source node waits for an acknowledgement to arrive. If the acknowledgement arrives, then the path is safe.
- If the acknowledgement does not arrive, the address of the last node that has a path to the destination node is stored in the suspect table, and retransmit the data using the second received path; go to steps (a, b).
- The nodes will exchange their suspect tables, in case of a common node is found in the exchanged lists, and the node is already in the suspect table, then it is moved to the black list table.
- Once a node is added to the black list table, RREP messages from this node are ignored.

Figure 2 depicts the procedures of the proposed algorithm that to solve a black hole problem in a consistent and sequential manner.

```
ST: Suspect Table (ST)
BLT:  Black List Table (BLT)
--------------------------------------------------------------------------------
Step 1: A RREQ packet is broadcasted by node i, wait a RREP then send data packet, and wait an Ack packet from destination
    While (no Acknowledgement is received)
        Increment the suspect probability for node i
        If ($P_{i,j}$ >= $P_{suspect}$)
            Insert replying address in suspect table
            Resend data to other RREP

Step 2: Broadcast ST
Step 3: While (received ST is not empty)
            if( $P_{i,j}$ >= $P_{black\ hole}$)
                        Move node j from ST to BLT
            otherwise
                        Rebroadcast Received ST
Step 4: Broadcast BLT
Step 5: While (received BLT is not empty)
                        Insert in this BLT
                        Broadcast this BLT
```

Figure 2. Proposed AODV Protocol Algorithm.

IV. Simulation and Analysis of Results

In this study, we use GloMoSim simulator [26] to evaluate the performance of three deferent protocols: proposed protocol, original AODV protocol, and MI-AODV protocol.

In order to evaluate the performance of the proposed scheme, different experiments with different number of nodes, namely, 15, 20, 25, 30, and 35 nodes, were conducted. The nodes placed randomly and move according to the random waypoint model with a speed of (0 – 20 m/s) over a square terrain area of 1000*1000 meters. Each run lasts for 800 seconds. The radio propagation range is 250 meters, and the bandwidth is 2 Mb/s. In the application layer, the Constant Bit Rate (CBR) traffic generator is used as a model of data resources in the simulations and the size of each data packet is 512 byte. In the MAC layer (i.e., Data Link Layer), we used the IEEE 802.11 communication protocol. Table 1 shows the simulation parameters for the different scenarios.

TABLE I. SIMULATION PARAMETERS

Parameter	Value
Simulator	GloMoSim 2.03
Simulation time	800 second
Simulation area	1000m × 1000m
Number of nodes	15, 20, 25, 30, and 35,
Mobility model	Random waypoint
Minimum speed	0 meter/second
Maximum speed	20 meter/second
Pause time	0 ,
MAC protocol	IEEE 802.11
Data packet size	512 byte
Radio range	250m
Bandwidth	2 Mb/s

The simulation evaluates the performance of the original AODV, MI-AODV and the proposed versions of AODV with the presence of 1, 2, 6 black hole nodes for each protocol. Each experiment was repeated 10 times with different random seeds to change the random simulator parameters; the average of the obtained 10 values is computed. The margin of error for each average at 95% confidence is computed. Four performance metrics were used in this study to evaluate and compare the proposed AODV to the MI-AODV and the original one. These metrics are: packet delivery ratio, dropped packets ratio, overhead, and end-to-end delay.

A. Results and Analysis

In this section, we provide analysis of the results obtained from the simulation experiment that we performed to compare the performance of the three protocols in the presence of the black hole nodes. Throughout the paper, the green line with the triangular markers represents the original

ISBN: 978-1-61208-339-1

AODV protocol, the red line with the square markers represents the MI-AODV protocol, and the blue line with the trapezoidal markers represents the proposed protocol.

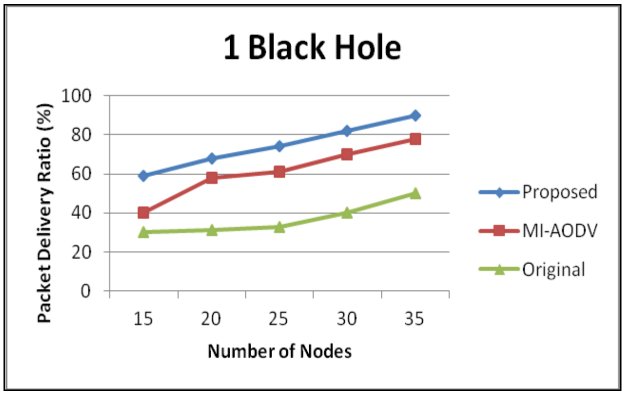


Figure 3. Delivery ratio, 1 Black hole, Pause 0.

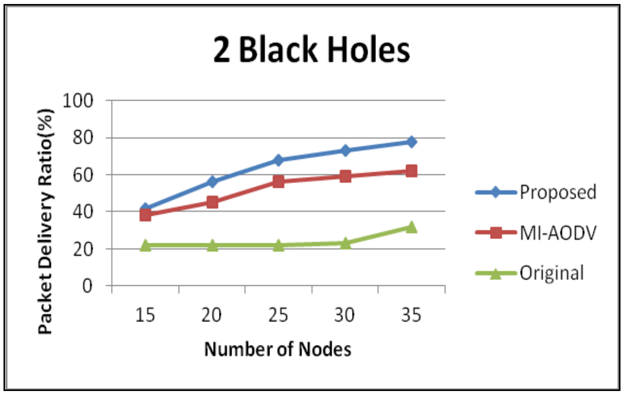


Figure 4. Delivery ratio, 2 Black holes, Pause 0.

Figure 3 shows the improvement of packets delivery ratio in the proposed AODV protocol compared to MI-AODV and original AODV protocols when the network is attacked by one black hole. Figure 4 shows the improvement of packets delivery ratio as the network is being attacked by two black hole nodes. As shown in Figures 3 and 4, the proposed AODV protocol improves the delivery ratio by 50.9% in case of 1 black hole and by 57.8% in case of 2 black holes; the MI-AODV protocol improves the delivery ratio by 38.4% in case of 1 black hole and by 48.5 in case of 2 black holes, compared to the original AODV protocol for a network attacked by one and two black hole.

Figures 3 and 4 show the results of a network attacked by one and two black holes, the packets delivery ratio for the cases of 15 to 35 nodes increases as the number of nodes increases. Within this interval, as the number of nodes decreases the effect of black hole increases, because a black hole has the chance to obtain more RREQ messages from all RREQ messages sent in the network; therefore, it drops more packets. This is the reason behind the decreasing packets delivery ratio for all protocols when the number of nodes decreases.

The packets delivery ratio increases as the number of nodes increases from 15 to 35 nodes for the original AODV, the MI-AODV, and the proposed AODV protocols. As the number of nodes increases within this interval a black hole has the chance to subscribe in more communications; however, the source node surrounded by more neighbor nodes therefore it has a greater chance to receive routes from other normal and reliable nodes.

Figures 5 and 6 show the ratio of dropped packets results for the three protocols for a network attacked by one and two black hole. The proposed AODV protocol improves the dropped packets ratio by 61.5% and 57.8% for the cases of 1 and 2 black holes respectively compared to the original AODV protocol. The MI-AODV protocol improves the dropped packets ratio by 39.7% and 48.5% for the cases of 1 and 2 black holes respectively compared to the original AODV protocol.

The proposed AODV protocol reduces the ratio of dropped packets compared to the MI-AODV and the original AODV protocols for a network attacked by one black hole. As shown in Figure 5, the ratio of dropped packets increases as the number of nodes decreases for the cases of 15 to 35 nodes. As number of nodes decreases within this interval, the black hole has the chance to drop high ratio of sent packets. This is the reason behind the increasing ratio of dropped packets.

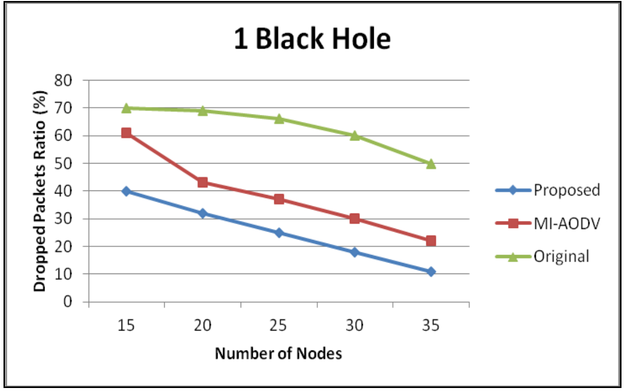


Figure 5. Dropped packets ratio, 1 Black holes, Pause 0.

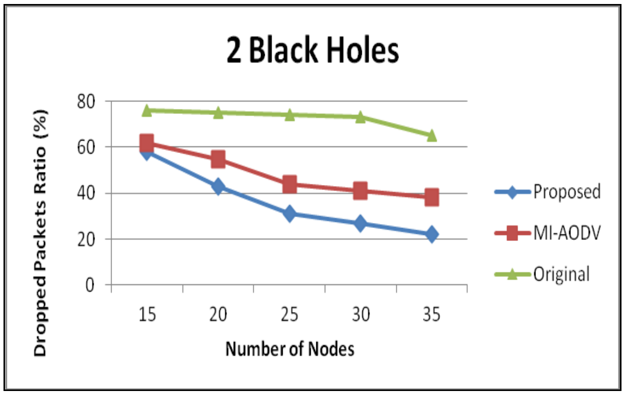


Figure 6. Dropped packets ratio, 2 Black holes.

When the number of nodes increases the source node becomes surrounded by more neighbors and has a high chance to receive more alternative routes to the desired destination and the effect of black hole nodes decreases. There is an observable agreement between the results of dropped packets ratio and delivery packets ratio for a network attacked by one and two black hole nodes. The obtained results for end-to-end delay show that the delay time is very close for the cases of 15 to 20 nodes because of the decreased number of nodes. This leads to increasing the chance of destination node to be neighbor to the source node. The original AODV protocol shows the best delay result compare to the proposed protocol by 24.3% and MI-AODV protocol by 11.6% for the case of one black hole.

Figure 7 depicts the results for delay times. The results indicate that by increasing the number of nodes, the delay increases for all protocols. Moreover, the original AODV achieves the lowest delay, while the proposed scheme achieves the highest delay; the increase in delay for the proposed scheme is due to the extra processing and resend of packets over the second discovered path to the destination. Therefore, the packet deliver ratio achieved by the proposed scheme is higher than the other 2 schemes.

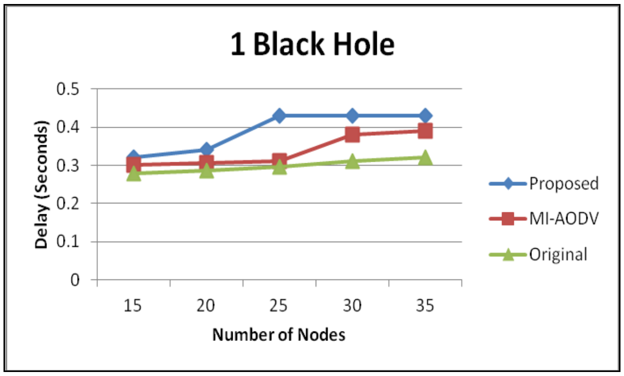


Figure 7. Delay, 1 Black hole.

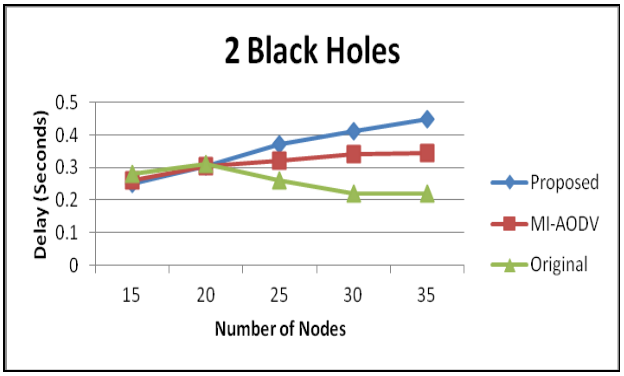


Figure 8. Delay, 2 Black hole.

In Figure 8, the network is attacked by two black holes and the delay time results are depicted. Similar behavior is depicted as in the case of 1 black hole except for the case of 15 to 20 nodes, in which the proposed scheme achieved the best results. For the case of 35 nodes, the original AODV protocol outperforms the proposed protocol by 18.3% and the MI-AODV protocol by 13.2%. Figures 9 and 10 depict the overhead results for the 3 protocols for the cases of 1 and 2 black holes, respectively. As shown in Figures 9 and 10, the proposed AODV protocol improves the additional overhead by 15.7% for the case of 1 black hole, and 15.1% for the case of 2 black holes, and the MI-AODV protocol improves the overhead by 6.9% for the case of 1 black hole, and 10.7% for the case of 2 black holes. The overhead reported by the original protocol is higher than the overhead reported by the other 2 protocols, while the proposed protocol achieved the lowest overhead.

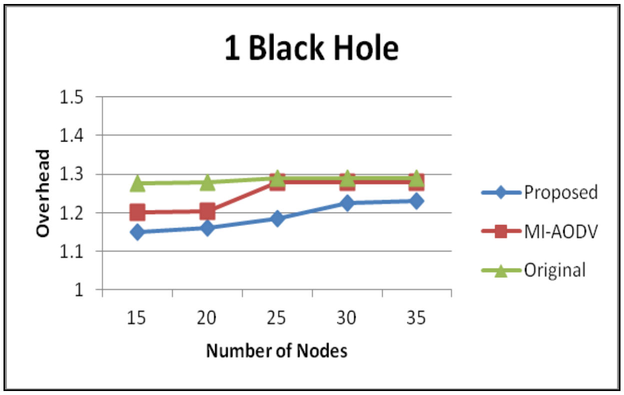


Figure 9. Overhead, 1 Black holes, Pause 0.

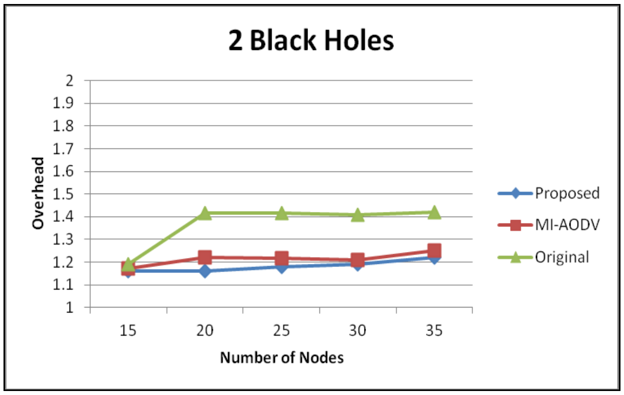


Figure 10. Overhead, 2 Black hole, Pause 0.

V. Conclusion and Future Work

The main focus of this research is security issue in MANETs because it is essential and even more challenging as it has multiple senders, multiple receivers, and the usage of wireless links for transmitting data. Black hole problem is type of denial of service attack where a malicious node can attract data packets by falsely advertise a fresh route to the destination and retain them without forwarding them to the destination. The proposed AODV protocol modify the behavior of the original AODV to send the data packets safely, and it aims at detecting and avoiding black hole nodes in MANET to reduce the impacts of black hole nodes.

This is due to the fast response of the black hole in the original scheme to the RREQs. This leads to increase the number RREQ and RREP control messages in the network. The transmission data processes by both MI-AODV and proposed protocols needs more time than the original AODV protocol, thus, the number of the control packets in MI-AODV and proposed AODV protocols is less than the number of control packets in the original AODV protocol.

Each node has suspect and black list tables to hold the addresses of the suspicions nodes, Suspect table contains the addresses of intermediate nodes which have sent RREP message, it also includes the number of times a node failed to send data through this node, and black list table contains a list of nodes with failed RREP message that exceeded a certain threshold. RREP is overloaded with an extra field to store address of the last node reply has a path to the destination. We added a new acknowledgment message to acknowledge the recipient of the send data from the source to the destination nodes. The obtained simulation results shown that the proposed AODV protocol comprehend the ill effects of the black hole attack and outperforms both the MI-AODV, and original AODV protocols in terms of packet delivery ratio, dropped packets ratio, and overhead.

The protocol does not consider the behavior of two black hole nodes that cooperate together and work as a team. The next step is to support the protocol with a certain technique to solve the problem for more than one black hole cooperate together, and support it with a certain mechanism to deal with spoofing and reply acknowledgment from black hole.

REFERENCES

[1] R. Rangara, R. Jaipuria, G. Yenugwar, and P. Jawandhiya, “Intelligent Secure Routing Model for MANET”. Proceedings of Computer Science and Information Technology (ICCSIT), IEEE, vol. 3, pp. 452 - 456, 2010.

[2] J. Sen, S. Koilakonda, and A. Ukil, “A Mechanism for Detection of Cooperative Black Hole Attack in Mobile Ad Hoc Networks”, Proceedings of Intelligent Systems, Modelling and Simulation (ISMS), IEEE, pp. 338 - 343, 2011.

[3] P. Tsou, J. Chang, Y. Lin, H. Chao, and J. Chen, “Developing a BDSR Scheme to Avoid Black Hole Attack Based on Proactive and Reactive Architecture in MANETs”, Proceedings of Advanced Communication Technology (ICACT), IEEE, pp. 755 – 760, 2011.

[4] M. Medadian, A. Mebadi, and E. Shahri, “Combat with Black Hole Attack in AODV Routing Protocol”, Proceedings of First Asian Himalayas, IEEE, pp. 530 - 535, 2009.

[5] S. Lu, L. Li, K. Lam, and L. Jia, “SAODV: A MANET Routing Protocol that can Withstand Black Hole Attack”, Proceedings of Computational Intelligence and Security, vol. 2, pp. 421 - 425, 2009.

[6] S. Umang, B. Reddy, and M Hoda, “Enhanced intrusion detection system for malicious node detection in ad hoc routing protocols using minimal energy consumption”, In ITE journal, vol. 4, 2010, pp. 2084 - 2094.

[7] S. Kannan, T. Maragatham, S. Karthik, and V. Arunachalam, “A Study of attacks, Attack Detection and Prevention Methods in Proactive and Reactive Routing Protocols”, In Medwell journal, vol. 5, 2011, pp. 178-183.

[8] Y. Khamayseh, A. Bader, W. Mardini, and Muneer BaniYasein, “A New Protocol for Detecting Black Hole Nodes in Ad Hoc Networks”, In International Journal of Communication Networks and Information Security (IJCNIS), vol. 3, 2011, pp. 36-47.

[9] N. Bhalaji and A. Shanmugam, “A Trust Based Model to Mitigate Black Hole Attacks in DSR Based Manet”, In European Journal of Scientific Research, vol. 50 no. 1, 2011, pp. 6-15.

[10] A. Sangi, J. Liu, L. Zou, “A Performance Analysis of AODV Routing Protocol under Combined Byzantine Attacks in MANETs”, Proceedings of Computational Intelligence and Software Engineering CiSE , IEEE, pp. 1 - 5, 2009.

[11] D. Mishra, Y. Jain, S. Agrawal, “Behavior Analysis of Malicious Node in the Different Routing Algorithms in Mobile Ad Hoc Network (MANET)”, Proceedings of Advances in Computing, Control, & Telecommunication Technologies, pp. 621-623, 2009.

[12] T. Manikandan and K. Sathyasheela, “Detection Of Malicious Nodes in MANETs”, Proceedings of Communication Control and Computing Technologies (ICCCCT), IEEE, pp. 788 - 793, 2010.

[13] W. Gong, Z. You, D. Chen, X. Zhao, M. Gu, and K. Lam, “Trust Based Malicious Nodes Detection in MANET“, Proceedings of E-Business and Information System Security, IEEE, pp. 1 - 4, 2009.

[14] N. Bhalaji and A. Shanmugam, “Association Between Nodes to Combat Blackhole Attack in DSR Based MANET”, Proceeding of Wireless and Optical Communications Networks, pp. 1-5, 2009.

[15] L. Tamilselvan and V. Sankaranarayanan, “Prevention of Blackhole Attack in MANET”, Proceeding of Wireless Broadband and Ultra Wideband Communications, IEEE, pp. 21, 2007.

[16] A. Saini and H. Kumar, “Effect Of Black Hole Attack On AODV Routing Protocol In MANET”, In International Journal of Computer Science and Technology, vol. 1, 2010, pp. 1 – 4,.

[17] E. Gerhards-Padilla, N. Aschenbruck, P. Martini, M. Jahnke, and J. T¨olle, “A Detecting Black Hole Attacks in Tactical MANETs using Topology Graphs”, Proceeding of Local Computer Networks, IEEE, pp. 1043 - 1052, 2007.

[18] G. Mamatha and S. Sharma, “A New Combination Approach To Secure MANETS Against Attacks”, In International Journal of Wireless & Mobile Networks (IJWMN), vol. 2, 2010, pp.1-10.

[19] R. Das, B. Purkayastha, and P. Das, “Security Measures for Black Hole Attack in MANET: An Approach”, In International Journal of Engineering Science and Technology, vol. 3, 2011, pp. 2832- 2838.

[20] A. Sangi, J. Liu, and L. Zou, "A Performance Analysis of AODV Routing protocol under Combined Byzantine Attacks in MANETs" , International Conference on Computational Intelligence and Software Engineering,CiSE 2009, pp. 1-5, 2009.

[21] M. Medadian, M. Yektaie, and A. Rahmani "Combat with Black Hole Attack in AODV routing protocol in MANET", AH-ICI 2009. First Asian Himalayas International Conference on, pp. 3-5 Nov. 2009.

[22] Z. Min and Z. Jiliu, "Cooperative Black Hole Attack Prevention for Mobile Ad Hoc Networks", In Proceedings of the 2009 International Symposium on Information Engineering and Electronic Commerce. IEEE Computer Society, Washington, DC, USA, pp. 26-30, 2009.

[23] X. Zhang,Y. Sekiya, and Y, Wakahara, "Proposal of a Method to Detect BlackHole Attackin MANET", Autonomous Decentralized Systems, International Symposium on, pp. 23-25 March 2009

[24] S. Marti, T. Giuli, K. Lai, and M. Bake, “Mitigating Routing Misbehavior”. In Proceedings of Mobile Ad hoc networks 6th MobiCom, BA Massachuestts; pp. 10-18, 2000.

[25] N. Mistry, D. Jinwala, and M. Zaveri, "Improving AODV Protocol against Blackhole Attacks", proceedings of the International Multi Conference of Engineers and Computer Scientists, vol. 2, 2010.

[26] X. Zeng, R. Bagrodia, M. Gerla, "GloMoSim: a library for parallel simulation of large-scale wireless networks," Parallel and Distributed Simulation, 1998. PADS 98. Proceedings. Twelfth Workshop on , pp. 154-161, May 199

Building a Portable Talking Medicine Reminder for Visually Impaired Persons

Hsiao Ping Lee*† and Tzu-Fang Sheu*‡
*Department of Medical Informatics, Chung Shan Medical University, Taichung, Taiwan, 40201 ROC
Email: ping@csmu.edu.tw
†Department of Medical Research, Chung Shan Medical University Hospital, Taichung, Taiwan, 40201 ROC
‡Department of Computer Science and Communication Engineering, Providence University, Taichung, Taiwan, 43301 ROC
Email: fang@pu.edu.tw

***Abstract*—Assistive systems can help improving the ability of disabled persons in taking care of themselves. There are more than 39 million visually impaired persons in the world, and moreover, the amount is increasing year by year. Therefore, Developing assistive systems for the visually impaired persons is an important issue. Since the medication guides are often too complex for visually impaired persons, it makes The visually impaired persons frequently forgetting to take drugs or being wrong in medication. In this work, we are going to develop a portable talking medicine reminding system to assist visually impaired persons in medication and reduce the number of medication errors. The assistive system will mainly provide auditory feedback in operation, and braille output for special needs. The assistive system will also provide a user-friendly interface that is specifically designed for visually impaired persons and is compatible with common-used screenreaders. The assistive system identifies drugs based on the now relatively common barcodes on drug begs. In addition, to overcome the problem that it is difficult for visually impaired persons to scan the barcode horizontally or vertically, an omni-directional barcode recognition technology will be used in the assistive system. It will make the assistive system easier for visually impaired persons to use. The assistive system will be built on a handheld device for better portability so that it can assist visually impaired persons anytime and anywhere, and reduce the number of medication errors. The system will be suitable to be used in the applications of ubiquitous health-care.**

***Keywords*—assistive systems for visually impaired persons; portable and talking assistive systems; medicine-taking reminding; barcode recognition; ubiquitous health-care.**

I. MOTIVATION

Assistive technology can help improve the ability of disabled persons in taking care of themselves. Building barrier-free smart living space and providing sufficient assistive systems and devices for disabled persons are one of the civilization indicators of a country [1]. The amount of visually impaired persons in the world is over 39 million [2]. The amount is increasing year by year. Due to their impairment, the visually impaired persons face more difficulty in their daily life. Most of them rely on assistive systems and devices to overcome the daily difficulties. There have been many assistive systems and devices developed for visually impaired persons in the past [3]–[10]. However, the amount and categories of assistive systems and devices for visually impaired persons are insufficient. The consequences are severe. It is important to develop appropriate assistive systems and devices, based on the needs of the visually impaired persons, to help solve some of the problems that they face in daily life, thereby improving their ability to take care of themselves.

Compared with sighted persons, visually impaired persons frequently forget to take drugs or make errors in medication because medication guides are often too complex for visually impaired persons, which include the category, dosage and taking time. Many visually impaired persons may make errors in the daily medication. The medication errors will reduce the quality of life of the visually impaired persons, and furthermore, may seriously damage their health. Currently, some medication reminding softwares have been developed and available in markets, for example, RxMindMe, Med Minder and OnTimeRx [11]–[13]. However, these softwares are designed for sighted persons rather than visually impaired persons. The special need of visually impaired persons are not considered in the design of the softwares. Thereby, it is worthwhile and important to develop an assistive system specific for visually impaired persons to assist them in medication and reduce the number of medication errors.

II. THE PORTABLE TALKING MEDICINE REMINDER FOR VISUALLY IMPAIRED PERSONS

In order to assist visually impaired persons in medication to reduce the number of medication errors, in this work, we are going to develop a portable assistive system for visually impaired persons, that is named Portable Talking Medicine Reminder (PTMR).

The developing PTMR system is built on handheld devices, for example Android smart phones, for better portability. Since barcodes are now relatively widely attached to drug begs, we will use barcode-based drug identification in the PTMR system so that the system knows the category of drugs in the beg. Due to their impairment, it is difficult for visually impaired persons to put barcodes under the camera len horizontally or vertically during recognizing barcodes. It will significantly impact the result of barcode recognition. Therefore, an omni-directional barcode recognition technology will be used in the system, that can correctly and efficiently recognize the barcode in any direction, for example a rotated barcode. The text-to-speech technology will be used in the PTMR system for auditory feedback. Also, the braille feedback will be available in the PTMR system as an optional output format. That is, the PTMR system is able to simultaneously provide messages in both auditory and braille formats. For convenient use, we are going to design a user-friendly interface specific designed for visually impaired persons. The user interface will be compatible with commonly used screenreading softwares. When users scan the barcode on a drug beg, the PTMR system recognizes the barcode, and then provides the medication information, such as drug name, taking time, dosage. The PTMR system also describes the color and the shape of the drug to avoid misery. The full guides in auditory and braille formats will be provided while the PTMR system is operated. In addition, we will build a service platform to achieve automatic collection of medication information in the work. A medication data exchange standard will be also designed for the use of transferring medication information between the user's client and service platform. The PTMR system can be configured automatically based on the medication information received from the service platform. The automatic configuration is one of the featured functions that facilitate the use of the PTMR system. It is difficult for visually impaired persons to input data on a handheld device with a touch screen. By using the featured function of automatic configuration, the problem of being unable to input data can be completely solved. Furthermore, the potential problem of typing error will be also avoided. The PTMR system will provide reminding and guiding functions to prevent visually impaired persons from medication errors, such as wrong type, time and dosage. The PTMR system will regularly send the medication and error logs to the visually impaired persons' nursing staffs or relatives automatically. Based on the logs, the nursing staffs or relatives can give necessary suggestions and care to the visually impaired persons in the medication. The portable talking medicine reminder developing in the work will serve visually impaired persons anytime and anywhere, and help to reduce the number of errors in medication.

III. Conclusion And Future Work

In this work, the assistive system that is called portable talking medicine reminder (PTMR) will be developed for visually impaired persons. The assistive system is designed to remind visually impaired persons in medication. The system helps avoid the medication errors, for example forgetting to take drugs or taking wrong drugs, and provide a better health-care. The PTMR system will provide several featured functions, such as user-friendly interfaces, auditory and braille feedback, automatic configuration, that facilitate the use of visually impaired persons. The PTMR system is portable and can be used anytime and anywhere. Furthermore, the PTMR system can be used in the applications of ubiquitous health-care. The work is still in progress.

Acknowledgment

The authors would like to thank the National Science Council of the Republic of China, Taiwan, for financially supporting this research under Grants [NSC102-2218-E-040-001 to H.P. Lee, NSC102-2221-E-040-004 to H.P. Lee].

References

[1] C. Phillips and M. Giasolli, “Assistive technology enhancement using human factors engineering,” in *Proceedings of the 1997 Sixteenth Southern Biomedical Engineering Conference*, 1997, pp. 26–29.

[2] “Prevalence of vision impairment,” http://www.lighthouse.org/research/statistics-on-vision-impairment/prevalence-of-vision-impairment/, accessed May, 2014.

[3] D. J. Calder, “Assistive technology interfaces for the blind,” in *Proceedings of the 3rd IEEE International Conference on Digital Ecosystems and Technologies*, 2009, pp. 318–323.

[4] H. Takizawa, S. Yamaguchi, M. Aoyagi, N. Ezaki, and S. Mizuno, “Kinect cane: An assistive system for the visually impaired based on three-dimensional object recognition,” in *Proceedings of IEEE/SICE International Symposium on System Integration (SII)*, 2012, pp. 740–745.

[5] J. Xiao, K. Ramdath, M. Iosilevish, D. Sigh, and A. Tsakas, “A low cost outdoor assistive navigation system for blind people,” in *Proceedings of the 8th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2013, pp. 828–833.

[6] T. H. Nguyen, T. H. Nguyen, T. L. Le, T. T. H. Tran, N. Vuillerme, and T. P. Vuong, “A wireless assistive device for visually-impaired persons using tongue electrotactile system,” in *Proceedings of 2013 International Conference on Advanced Technologies for Communications (ATC)*, 2013, pp. 586–591.

[7] R. Velzquez, *Wearable and Autonomous Biomedical Devices and Systems for Smart Environment*. Springer Berlin Heidelberg, 2010.

[8] D. J. Calder, “Assistive technologies and the visually impaired: a digital ecosystem perspective,” in *Proceedings of the 3rd International Conference on PErvasive Technologies Related to Assistive Environments*, 2010.

[9] D. Dakopoulos and N. G. Bourbakis, “Wearable obstacle avoidance electronic travel aids for blind: a survey,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 40, no. 1, pp. 25–35, 2010.

[10] S. Shoval, J. Borenstein, and Y. Koren, “The navbelt-a computerized travel aid for the blind based on mobile robotics technology,” *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 11, pp. 1376–1386, 1998.

[11] “Rxmindme,” https://itunes.apple.com/us/app/rxmindme-prescription-medicine/id379864173?mt=8, accessed May, 2014.

[12] “Medreminder,” https://play.google.com/store/apps/details?id=com.rsys&hl=zh_TW, accessed May, 2014.

[13] “Ontimerx,” http://www.ontimerx.com/, accessed May, 2014.

ISBN: 978-1-61208-339-1

Physarum Syllogistic L-Systems

Andrew Schumann
University of Information Technology and Management
Rzeszow, Poland
e-mail: Andrew.Schumann@gmail.com

***Abstract*—One of the best media for studying natural computing is presented by the behavior of *Physarum Polycephalum* plasmodia. Plasmodium has active zones of growing pseudopodia and these zones interact concurrently and in a parallel manner. This behavior can be stimulated by attractants and repellents. In the paper, different syllogistic systems are proposed for simulating the plasmodium's behavior. While Aristotelian syllogistic may describe concrete directions of *Physarum* spatial expansions, pragmatic syllogistic proposed in this paper may describe *Physarum* simultaneous propagations in all directions. It is a more suitable system for applying syllogistic models in designing logic gates in plasmodia.**

Keywords-plasmodium; L-system; Aristotelian syllogistics; non-Aristotelian syllogistics.

I. INTRODUCTION

There are many approaches to biological computing as a kind of unconventional computing; one of them is presented by systems invented by Aristid Lindenmayer [4]. They are called L-systems and allow us to simulate the growth of plants by formal grammars [6][13]. In the project [2], we are going to develop another approach to biological computing, assuming a massive parallelism of biological behavior. In this paper, we will show that we can implement two syllogistics in the biological behavior: the Aristotelian syllogistic [5] and a non-Aristotelian syllogistic constructed in [10][12]. The first is implementable within standard trees of appropriate L-systems. The second is massive-parallel and contain cycles and, therefore, can be implementable just within non-standard trees of some rigorous extensions of L-systems. This means that *Physarum Polycephalum*, the medium of computations, which we have studied in the project, embodies complex extensions of L-systems.

Let us recall that *Physarum Polycephalum* is a one-cell organism that behaves according to different stimuli and can be considered the basic medium of simple actions that are intelligent in the human meaning [1][2][7][8][9]. It behaves by plasmodia which can have the form either waves or protoplasmic tubes (arches). Hence, it is a system that is being spatially extended, as well as standard L-systems. This extension can be described as an extension of L-system called *Physarum* L-system (Section 2). Within this system we can implement (i) Aristotelian syllogistic in the *Physarum* media (Section 4), as well as non-Aristotelian syllogistic defined in [10][12] (Section 5).

In our project [2], we obtained a basis of new object-oriented programming language for *Physarum polycephalum* computing [11]. Within this language we are going to check possibilities of practical implementations of storage modification machines on plasmodia and their applications to behavioral science such as behavioral economics and game theory. The point is that experiments with plasmodia may show fundamental properties of any intelligent behavior. The language, proposed by us, can be used for developing programs for *Physarum Polycephalum* by the spatial configuration of stationary nodes. Some preliminary results of computational models on plasmodia are obtained in [1]. In this paper, we consider possibilities to implement syllogistic models as logic gates for *Physarum Polycephalum*, which can be programmable within our language. In Section 2, we define *Physarum* L-systems. In Section 3, we consider their particular case presented by Aristotelian trees. In Section 4, we show how we can implement Aristotelian syllogistic in the *Physarum* behavior. In Section 5, we show how we can implement non-Aristotelian syllogistic defined in [10][12].

II. PHYSARUM L-SYSTEM

The behavior of *Physarum* plasmodia can be stimulated by attractants and repellents. We have the following entities which can be used in programming plasmodia:

- The set of *active zones* of *Physarum* $\{V_1, V_2, ...\}$, from which any behavior begin to carry out.
- The set of *attractants* $\{A_1, A_2, ...\}$; they are sources of nutrients, on which the plasmodium feeds, or pheromones which chemically attract the plasmodium. Any attractant is characterized by its position and intensity.
- The set of *repellents* $\{R_1, R_2, ...\}$. Plasmodium of *Physarum* avoids light and some thermo- and salt-based conditions. Thus, domains of high illumination (or high grade of salt) are repellents such that each repellent is characterized by its position and intensity, or force of repelling.
- The set of *protoplasmic tubes* $\{T_1, T_2, ...\}$. Typically, plasmodium spans sources of nutrients with protoplasmic tubes/veins. The plasmodium builds a planar graph, where nodes are sources of nutrients or pheromones, e.g., oat flakes, and edges are protoplasmic tubes.

Plasmodia grow from active zones. At these active zones, according to Adamatzky's experiments [2][3], the following three basic operations stimulated by nutrients (attractants) and some other conditions can be observed: fusion, multiplication, and direction operations (see Fig. 1):

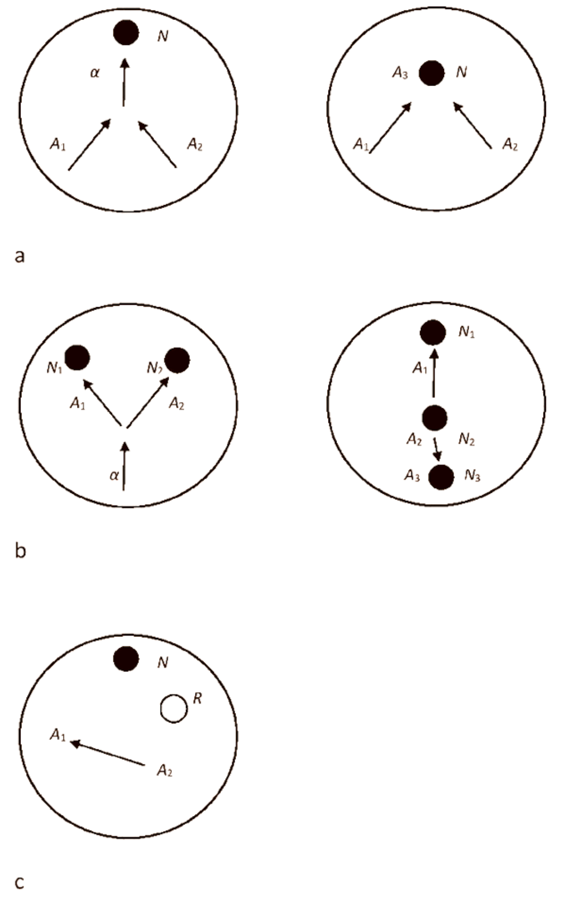


Figure 1. The stimulation of the following operations in Physarum automata: (a) fusion, (b) multiplication, and (c) direction, where A_1, A_2, A_3 are active zones, N, N_1, N_2, N_3 are attractants, α is a protoplasmic tube, R is a repellent.

(1) The *fusion*, denoted *Fuse*, means that two active zones A_1 and A_2 either produce new active zone A_3 (i.e., there is a collision of the active zones) or just a protoplasmic tube α: $Fuse(A_1, A_2) = A_3$ or $Fuse(A_1, A_2) = \alpha$.

(2) The *multiplication*, *Mult*, means that the active zone A_1 splits into two independent active zones A_2 and A_3, propagating along their own trajectories: $Mult(A_1) = \{A_2, A_3\}$ or $Mult(\alpha) = \{A_2, A_3\}$.

(3) The *direction*, *Direct*, means that the active zone A is not translated to a source of nutrients but to a domain of an active space with certain initial velocity vector v: $Direct(A, v)$.

These operations, *Fuse*, *Mult*, *Direct*, can be determined by the attractants $\{A_1, A_2, ...\}$ and repellents $\{R_1, R_2, ...\}$.

On the basis of active zones $\{V_1, V_2, ...\}$, attractants $\{A_1, A_2, ...\}$, repellents $\{R_1, R_2, ...\}$, and protoplasmic tubes $\{T_1, T_2, ...\}$, we can define a *Physarum* L-system. Let us remember that an L-system consists of (i) an alphabet of symbols that can be used to make strings, (ii) a collection of production rules that expand each symbol into some larger or shorter string of symbols, and (iii) an initial string from which we move. These systems were introduced by Lindenmayer [4][6][13] to describe and simulate the behavior of plant cells.

The *Physarum* L-system is defined as follows: $\mathbf{G} = \langle G, \omega, Q\rangle$, where (i) G (the *alphabet*) is a set of symbols containing elements that can be replaced (*variables*), namely they are active zones $\{V_1, V_2, ...\}$, which can be propagated towards attractants $\{A_1, A_2, ...\}$ by protoplasmic tubes and avoid repellents $\{R_1, R_2, ...\}$, i.e., $G = \{V_1, V_2, ...\} \cup \{A_1, A_2, ...\} \cup \{R_1, R_2, ...\}$; (ii) ω (*start*, *axiom* or *initiator*) is a string of symbols from G defining the initial state of the system, i.e., ω always belongs to $\{V_1, V_2, ...\}$; (iii) Q is a set of *production rules* or *productions* defining the way variables can be replaced with combinations of constants and other variables, i.e., production rules show a propagation of active zones by protoplasmic tubes towards attractants with avoiding repellents.

Let A, B, C are called primary strings, their meanings run over symbols V_1, V_2, ..., A_1, A_2, ... Production rules allow us to build composite strings from primary strings. So, a production A $\rightarrow_Q$ B consists of two strings, the *predecessor* A and the *successor* B. Some basic cases of productions are as follows: (i) the *fusion*, denoted AB $\rightarrow_Q$ C, means that two active zones A and B produce new active zone C at the place of an attractant denoted by C; (ii) the *multiplication*, A $\rightarrow_Q$ BC, means that the active zone A splits into two independent active zones B and C propagating along their own trajectories towards two different attractants denoted then by B and C; (iii) the *direction*, A $\rightarrow_Q$ B, means that the active zone A is translated to a source of nutrients B.

L-systems can generate infinite data structure. Therefore it is better to define some production rules, denoted by A → B, recursively like that: A → BA, producing an infinite sequence BABABABA… from A, or A → BCA, producing an infinite sequence BCABCABCABCA… from A. In the *Physarum* L-system, the rule A → BA means that we will fulfill the *direction*, A $\rightarrow_Q$ B, infinitely many time, the rule A → BCA means that we will fulfill the *multiplication*, A $\rightarrow_Q$ BC, infinitely many time. Let us consider an example of recursive production rules. Let G = {A, B} and let us start with the string A. Assume (A → BA) and (B → B). Thus, we obtain the following strings:

Generation $n = 0$: A
Generation $n = 1$: BA
Generation $n = 2$: BBA
Generation $n = 3$: BBBA
Generation $n = 4$: BBBBBA
Generation $n = 5$: BBBBBBA

In an appropriate *Physarum* L-system, these generations are represented as an infinite tree by permanent additions new attractants before the plasmodium propagation. In other words, we obtain the binary tree labeled with *s* and *t*, and whose interior nodes are either one unary node labeled with B or one binary node labeled with A (Fig. 2).

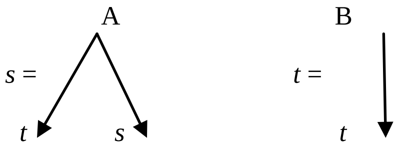


Figure 2. Example of labels for binary trees.

To sum up, we obtain the infinite binary tree of Fig. 3.

If we are limited just by the *multiplication*, A $\rightarrow_Q$ BC, and the *direction*, A $\rightarrow_Q$ B, we can build up binary trees in *Physarum* L-systems using the following definition of binary trees labeled with *x, y*, …, whose interior nodes are either unary nodes labeled with u_1, u_2, … or binary nodes labeled with b_1, b_2, …:

1. the variables *x*, *y*, … are trees;
2. if *t* is a tree, then adding a single node labeled with one of u_1, u_2, … as a new root with *t* as its only subtree gives a tree;
3. if *s* and *t* are trees, then adding a single node labeled with one of b_1, b_2, … as a new root with *s* as the left subtree and *t* as the right subtree again gives a tree;
4. trees may go on forever.

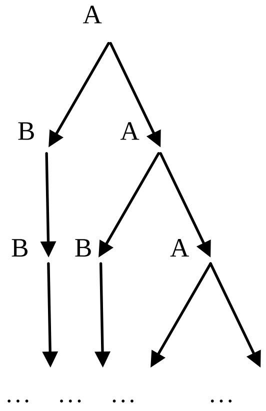


Figure 3. Example of infinite binary tree.

Let Tr be the set of trees that we have been defined. Then our definition introduces a coalgebra [14]:

$$\mathrm{Tr} = \{x, y, \ldots\} \cup (\{u_1, u_2, \ldots\} \times \mathrm{Tr}) \cup (\{b_1, b_2, \ldots\} \times \mathrm{Tr} \times \mathrm{Tr}).$$

Thus, within L-systems, we can obtain complex structures including infinite structures defined coalgebraically. In some cases, it is better to deal with infinite structures (infinite trees), assuming that all strings are finite.

III. Aristotelian Trees

Let us consider Aristotelian syllogistic trees, which can be large, but their strings are only of the length 1 or 2. An *Aristotelian syllogistic tree* is labeled with *x*, *y*, …, its interior nodes are *n*-ary nodes labeled with b_1, b_2, …, and it is defined as follows: (1) the variables *x*, *y*, … are Aristotelian syllogistic trees whose single descendants are underlying things (*hypokeimenon*, ὑποκείμενον) such that for each *x*, *y*, …, parents are supremums of descendants (notice that all underlying things are mutually disjoint); (2) if t_1, t_2, …, t_n are Aristotelian syllogistic trees such that their tops are concepts which are mutually disjoint and their supremum is $b_x \in \{b_1, b_2, \ldots\}$, then adding a single node labeled with b_x as a new root with t_1, t_2, …, t_n as its only subtrees gives an Aristotelian syllogistic tree; (3) an Aristotelian syllogistic tree is finite.

The idea of *hypokeimenon* allowed Aristotle to build up finite trees. He starts with underlying things as primary descendants of trees in constructing syllogistic databases. Now, let us define syllogistic strings of the length 1 or 2 by means a *Physarum* L-system. Let each $b_x \in \{b_1, b_2, \ldots\}$ be presented by an appropriate attractant and underlying things by initial active zones of Physarum. So, first trees *x*, *y*, …, whose single descendants are underlying things, are obtained by fusion or direction. Their supremums are denoted by attractants which were occupied by the first plasmodium propagation. These trees are considered subtrees for the next plasmodia propagation by fusion or direction. At the end, we can obtain just one supremum combining all subtrees. Let a_1, a_2, a_3,… be underlying things. Then they are initial strings, i.e., they can be identified with active zones of plasmodia. Their meanings are as follows: "there exists a_1", "there exists a_2", "there exists a_3", … Assume that in the tree structure the supremum of a_1 and a_2 is b_1, the supremum of a_2 and a_3 is b_2, … These supremums are fusions of plasmodia. Then, we have the strings a_1b_1, a_2b_1, a_2b_2, a_3b_2, … Their meanings are as follows: "a_1 is b_1", "a_2 is b_1", "a_2 is b_2", "a_3 is b_2", … Further, let b_n be a supremum for b_1 and b_2. It denotes an attractant that was occupied by the plasmodium at the third step of the propagation. Our new strings are as follows: "b_1 is b_n", "b_2 is b_n", etc. Now we can appeal also to the following new production rule: if "*x* is *y*" and "*y* is *z*", then "*x* is *z*". Thus, we have the strings: a_1b_n, a_2b_n, a_2b_n, a_3b_n, …

IV. Aristotelian Syllogistic

The symbolic system of Aristotelian syllogistic can be implemented in the behavior of *Physarum* plasmodium. Let us design cells of *Physarum* syllogistic which will designate classes of terms. We can suppose that cells can possess different topological properties. This depends on intensity of chemo-attractants and chemo-repellents. The intensity entails the natural or geographical neighborhood of the set's elements in accordance with the spreading of attractants or repellents. As a result, we obtain Voronoi cells [3][11]. Let us define what they are mathematically. Let **P** be a nonempty finite set of planar points and $|\mathbf{P}| = n$. For points $p = (p_1, p_2)$ and $x = (x_1, x_2)$, let

$$d(p,x) = \sqrt{(p_1 - x_1)^2 + (p_2 - x_2)^2}$$

denote their Euclidean distance. A planar Voronoi diagram of the set **P** is a partition of the plane into cells, such that for any element of **P**, a cell corresponding to a unique point p contains all those points of the plane which are closer to p in respect to the distance d than to any other node of **P**. A unique region

$$vor(p) = \bigcap_{m \in \mathbf{P}, m \neq p} \{z \in \mathbf{R}^2 : d(p,z) < d(m,z)\}$$

assigned to the point p is called a *Voronoi cell* of the point p. Within one Voronoi cell, a reagent has a full power to attract or repel the plasmodium. The distance d is defined by intensity of reagent spreading like in other chemical reactions simulated by Voronoi diagrams. A reagent attracts or repels the plasmodium and the distance on that it is possible corresponds to the elements of a given planar set **P**. When two spreading wave fronts of two reagents meet, this means that on the board of meeting the plasmodium cannot choose its one further direction and splits (see Fig. 5). Within the same Voronoi cell, two active zones will fuse.

Now, we can obtain coordinates $(x, y) \in \mathbf{Z}^2$ for each Voronoi center. The number (x, y) can be assigned to each concept as its character. If a Voronoi center with the coordinates (x_a, y_a) is presented by an attractant that is activated and occupied by the plasmodium, this means that in an appropriate *Physarum* syllogistic model there exists a string a with the coordinates (x_a, y_a). This string has the meaning "*a* exists". If a Voronoi center with the coordinates (x_a, y_a) is presented by a repellent that is activated and avoided by the plasmodium, this means that in an appropriate *Physarum* syllogistic model there exists a string $[a]$ with the coordinates (x_a, y_a). This string has the meaning "*a* does not exist". If two neighbor Voronoi cells with the coordinates (x_a, y_a) and (x_b, y_b) of centers contain activated attractants which are occupied by the plasmodium and between both centers there are protoplasmic tubes, then in an appropriate *Physarum* syllogistic model there exists a string ab and a string ba where a has the coordinates (x_a, y_a) and b has the coordinates (x_b, y_b). The meaning of those strings is the same and it is as follows: "*ab* exist", "*ba* exist", "some *a* is *b*", "some *b* is *a*".

If one neighbor Voronoi cell with the coordinates (x_a, y_a) of its center contains an activated attractant which is occupied by the plasmodium and another neighbor Voronoi cell with the coordinates (x_b, y_b) of its centre contains an activated repellent which is avoided by the plasmodium, then in an appropriate *Physarum* L-system there exists a string $a[b]$ and a string $[b]a$ where a has the character (x_a, y_a) and $[b]$ has the character (x_b, y_b). The meaning of those strings is the same and it is as follows: "*ab* do not exist, but *a* exists without *b*", "there exists *a* and no *a* is *b*", "no *b* is *a* and there exists *a*", "*a* exists and *b* does not exist".

If two neighbor Voronoi cells with the coordinates (x_a, y_a) and (x_b, y_b) of their centers contain activated repellents which are avoided by the plasmodium, then in an appropriate *Physarum* L-system there exists a string $[ab]$ and a string $[ba]$ where $[a]$ has the character (x_a, y_a) and $[b]$ has the character (x_b, y_b). The meaning of those strings is the same and it is as follows: "*ab* do not exist together", "there are no *a* and there are no *b*", "no *b* is *a*", "no *a* is *b*". Hence, existence propositions of Aristotelian syllogistic are spatially implemented in *Physarum* L-systems.

Let y' denote all neighbor Voronoi cells for x which differ from y. Now, let us consider a complex string $xy\&x[y']$. The sign & means that we have strings xy and $x[y']$ simultaneously and they are considered the one complex string. The meaning of the string $xy\&x[y']$ is a universal affirmative proposition "all *x* are *y*".

As a consequence, each *Physarum* L-system is considered a discourse universe verifying some propositions of Aristotelian syllogistic.

V. Non-Aristotelian Syllogistic

Let us propose now the syllogistic system formalizing performative propositions of the form '*A* is *P*' (see [10][12]), i.e., propositions with context-based meanings. This system is said to be *synthetic* (*pragmatic*) *syllogistic*, while we are assuming that Aristotelian syllogistic is analytic (informative). The basic logical connectives of pragmatic syllogistic are as follows: a ('every + noun + is + adjective'), i ('some + noun + is + adjective'), e ('no + noun + is + adjective') and o ('some + noun + is not + adjective') that are defined in the following way:

$$S\mathrm{a}P := \exists A\ (A \text{ is } S) \wedge (\forall A(A \text{ is } S \wedge A \text{ is } P)). \qquad (1)$$

$$S\mathrm{i}P := \forall A(\neg(A \text{ is } S) \wedge \neg(A \text{ is } P)). \qquad (2)$$

$$S\mathrm{o}P := \neg(\exists A\ (A \text{ is } S) \wedge (\forall A(A \text{ is } S \wedge A \text{ is } P))), \text{ i.e.,} \qquad (3)$$

$$\forall A \neg(A \text{ is } S) \vee (\exists A(\neg(A \text{ is } S) \vee \neg(A \text{ is } P))).$$

$$S\mathrm{e}P := \neg(\forall A(\neg(A \text{ is } S) \wedge \neg(A \text{ is } P))), \text{ i.e.,} \qquad (4)$$

$$\exists A(A \text{ is } S \vee A \text{ is } P).$$

Now, let us formulate axioms of pragmatic syllogistic:

$$S\mathrm{a}P \Rightarrow S\mathrm{e}P. \qquad (5)$$

$$S\mathrm{a}P \Rightarrow P\mathrm{a}S. \qquad (6)$$

$$S\mathrm{i}P \Rightarrow P\mathrm{i}S. \qquad (7)$$

$$S\mathrm{a}M \Rightarrow S\mathrm{e}P. \qquad (8)$$

$$M\mathrm{a}P \Rightarrow S\mathrm{e}P. \qquad (9)$$

$$(M\mathrm{a}P \wedge S\mathrm{a}M) \Rightarrow S\mathrm{a}P \qquad (10)$$

$$(M\mathrm{i}P \wedge S\mathrm{i}M) \Rightarrow S\mathrm{i}P. \qquad (11)$$

In pragmatic syllogistic, we have a novel square of opposition that we call the *synthetic square of opposition* (see Fig. 4), where the following theorems are inferred from (1) – (11):

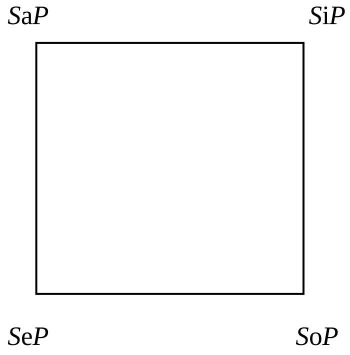


Figure 4. The synthetic square of opposition.

$S\mathrm{a}P \Rightarrow \neg(S\mathrm{o}P)$, $\neg(S\mathrm{o}P) \Rightarrow S\mathrm{a}P$, $S\mathrm{i}P \Rightarrow \neg(S\mathrm{e}P)$, $\neg(S\mathrm{e}P) \Rightarrow S\mathrm{i}P$, $S\mathrm{e}P \Rightarrow \neg(S\mathrm{i}P)$, $\neg(S\mathrm{i}P) \Rightarrow S\mathrm{e}P$, $S\mathrm{o}P \Rightarrow \neg(S\mathrm{a}P)$, $\neg(S\mathrm{a}P) \Rightarrow S\mathrm{o}P$, $S\mathrm{a}P \Rightarrow \neg(S\mathrm{i}P)$, $S\mathrm{i}P \Rightarrow \neg(S\mathrm{a}P)$, $\neg(S\mathrm{e}P) \Rightarrow S\mathrm{o}P$, $\neg(S\mathrm{o}P) \Rightarrow S\mathrm{e}P$, $S\mathrm{a}P \Rightarrow S\mathrm{e}P$, $S\mathrm{i}P \Rightarrow S\mathrm{o}P$, $S\mathrm{e}P \vee S\mathrm{i}P$, $\neg(S\mathrm{e}P \wedge S\mathrm{i}P)$, $S\mathrm{a}P \vee S\mathrm{o}P$, $\neg(S\mathrm{a}P \wedge S\mathrm{o}P)$, $\neg(S\mathrm{a}P \wedge S\mathrm{i}P)$, $S\mathrm{e}P \vee S\mathrm{o}P$.

For more details, see [10][12].

In the implementations within *Physarum* L-systems, the four basic syllogistic propositions of non-Aristotelian syllogistic defined above are understood as follows:

• 'All *S* are *P*': there is a string *AS* and for any *A* which is a neighbor for *S* and *P*, there are strings *AS* and *AP*. This means that we have a massive-parallel occupation of region, where the cells *S* and *P* are located.

• 'Some *S* are *P*': for any *A* which is a neighbor for *S* and *P*, there are no strings *AS* and *AP*. This means that the plasmodium cannot reach *S* from *P* or *P* from *S* immediately.

• 'No *S* are *P*': there exists *A* which is a neighbor for *S* and *P* such that there is a string *AS* or there is a string *AP*. This means that the plasmodium occupies *S* or *P*, but surely not the whole region, where the cells *S* and *P* are located.

• 'Some *S* are not *P*': for any *A* which is a neighbor for *S* and *P* there is no string *AS* or there exist *A* which is a neighbor for *S* and *P* such that there is no string *AS* or there is no string *AP*. This means that the plasmodium does not occupy *S* or there is a neighbor cell which is not connected with *S* or *P* by a protoplasmic tube.

Thus, the pragmatic syllogistic allows us to study different zones containing attractants for *Physarum* if they are connected by protoplasmic tubes homogenously.

VI. CONCLUSION

We constructed two syllogistic versions of storage modification machine in *Physarum Polycephalum*: Aristotelian syllogistic and pragmatic syllogistic (non-Aristotelian syllogistic of Section 5). While Aristotelian syllogistic may describe concrete directions of *Physarum* spatial expansions, pragmatic syllogistic may describe *Physarum* simultaneous propagations in all directions. Therefore, while for the implementation of Aristotelian syllogistic we need repellents to avoid some possibilities in the *Physarum* propagations, for the implementation of pragmatic syllogistic we do not need them. Hence, the second syllogistic can simulate massive-parallel behaviors, including different form of propagations such as processes of public opinion formation.

In our opinion, the general purpose of *Physarum* computing covers many behavioural sciences, because the slime mould's behaviour can be considered the simplest natural intelligent behaviour. Thus, our results may have an impact on computational models in behavioural sciences in general.

ACKNOWLEDGMENT

This research is supported by FP7-ICT-2011-8 and UMO-2012/07/B/HS1/00263.

REFERENCES

[1] A. Adamatzky, V. Erokhin, M. Grube, Th. Schubert, and A. Schumann, "Physarum Chip Project: Growing Computers From Slime Mould," International Journal of Unconventional Computing, 8(4), 2012, pp. 319-323.

[2] A. Adamatzky, "Physarum machine: implementation of a Kolmogorov-Uspensky machine on a biological substrate," Parallel Processing Letters, vol. 17, no. 04, 2007, pp. 455-467.

[3] A. Adamatzky, Physarum Machines: Computers from Slime Mould (World Scientific Series on Nonlinear Science, Series A). World Scientific Publishing Company, 2010.

[4] A. Lindenmayer, "Mathematical models for cellular interaction in development. parts I and II," Journal of Theoretical Biology, 18, 1968, pp. 280-299; 300-315.

[5] J. Łukasiewicz, Aristotle's Syllogistic From the Standpoint of Modern Formal Logic. Oxford Clarendon Press, 2nd edition, 1957.

[6] K. Niklas, Computer Simulated Plant Evolution. Scientific American, 1985.

[7] A. Schumann and A. Adamatzky, "Logical Modelling of Physarum Polycephalum," Analele Universitatii Din Timisoara, seria Matemtica-Informatica 48 (3), 2010, pp. 175-190.

[8] A. Schumann and A. Adamatzky, "Physarum Spatial Logic," New Mathematics and Natural Computation 7 (3), 2011, pp. 483-498.

[9] A. Schumann and L. Akimova, "Simulating of Schistosomatidae (Trematoda: Digenea) Behaviour by Physarum Spatial Logic," Annals of Computer Science and Information Systems, Volume 1. Proceedings of the 2013 Federated Conference on Computer Science and Information Systems. IEEE Xplore, 2013, pp. 225-230.

[10] A. Schumann, “On Two Squares of Opposition: the Leśniewski’s Style Formalization of Synthetic Propositions,” Acta Analytica 28, 2013, pp. 71-93.

[11] A. Schumann and K. Pancerz, “Towards an Object-Oriented Programming Language for Physarum Polycephalum Computing,” in M. Szczuka, L. Czaja, M. Kacprzak (eds.), Proceedings of the Workshop on Concurrency, Specification and Programming (CS&P'2013), Warsaw, Poland, September 25-27, 2013, pp. 389-397.

[12] A. Schumann, “Two Squares of Opposition: for Analytic and Synthetic Propositions,” Bulletin of the Section of Logic 40 (3/4), 2011, pp. 165-178.

[13] P. Prusinkiewicz and A. Lindenmayer, The Algorithmic Beauty of Plants. Springer-Verlag, 1990.

[14] J. J. M. M. Rutten, “Universal coalgebra: a theory of systems,” Theor. Comput. Sci., 249 (1), 2000, pp. 3-80.

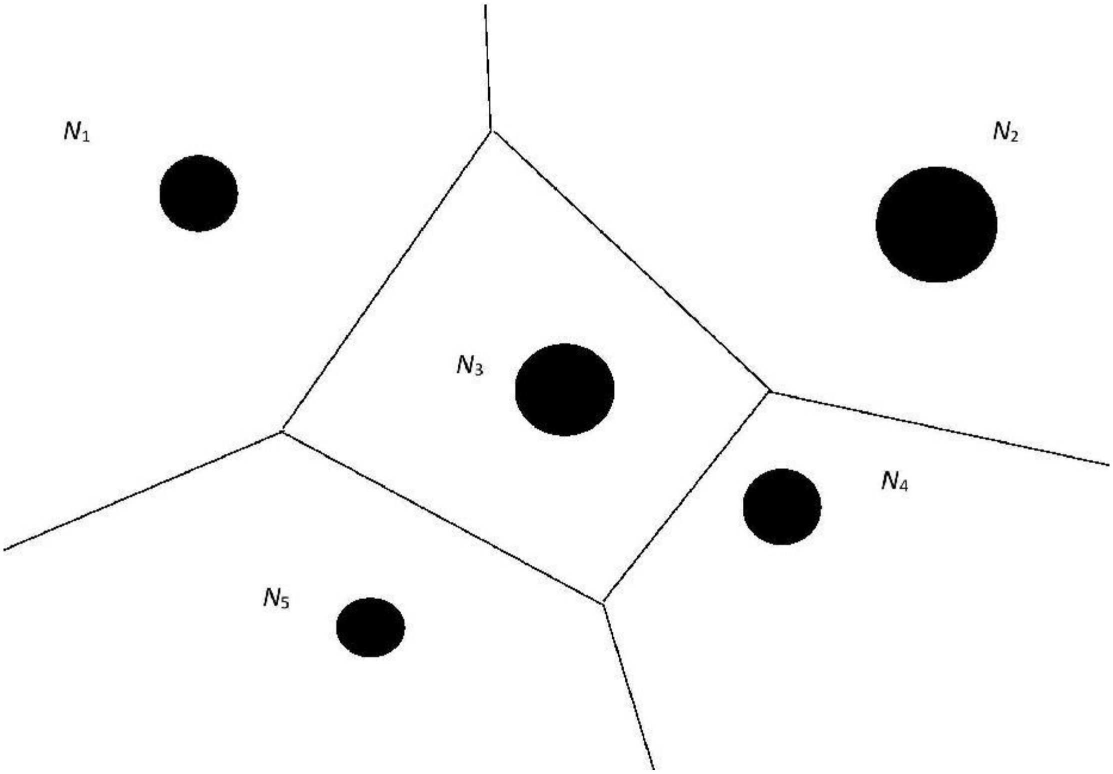


Figure 5. The Voronoi diagram for Physarum, where different attractants have different intensity and power.

Using Fuzzy Inference System to Estimate the Minimized Effective Dose and Critical Organ in Pediatric Nuclear Medicine

Ying Bai
Dept of Computer Science & Engineering
Johnson C. Smith University
Charlotte, NC USA
ybai@jcsu.edu

Dali Wang
Dept of Physics & Computer Science
Christopher Newport University
Newport News, VA, USA
dwang@pcs.cnu.edu

Abstract - **Many techniques and research models on calculating and reducing the nuclear radiation dose on pediatric nuclear medicine procedure have been developed and reported in recent years. However, most those models either utilized simple shapes to present the organs or used more realistic models to estimate the nuclear dose applied on pediatric patients. The former are too simple to provide accurate estimation results, and the latter are too complicated to intensively involve complex calculations. In this study, a simple but practical model is developed to enable physicians to easily and quickly calculate and select the average optimal effective nuclear dose and critical organs for the given age and weight of the pediatric patients. This model is built based on one research result reported by a group of researchers, and it can be easily implemented in most common pediatric nuclear medicine procedures. This is the first research of using fuzzy inference system to calculate the optimal effective dose applied in the nuclear medicine for pediatric patients.**

Keywords-fuzzy inference system; reduction of nuclear radiation dose; pediatric nuclear medicine; optimized nuclear radiation dose; nuclear medicine

I. Introduction

Nuclear medicine provides important and critical information that assists in the diagnosis, treatment, and follow-up of a variety of disorders on pediatric patients, including central nervous, endocrine, cardiopulmonary, renal, and gastrointestinal systems, as well as in the fields of oncology, orthopedics, organ transplantation, and surgery. Due to its high sensitivity, nuclear medicines can detect some disease in its earliest stages to enable it to be treated earlier. The noninvasive nature of nuclear medicine makes it an extremely valuable diagnostic tool for the evaluation of children. It provides useful diagnostic information that may not be easily obtained by using other diagnostic methods, some of them may be more invasive or contain some higher nuclear radiations [1][2].

Pediatric nuclear medicine includes the application of small amounts of radiopharmaceuticals that emit nuclear radiations such as γ-rays, β-particles, or positrons to patients during the diagnostic process. This emission exposes the pediatric patients to low levels of nuclear radiations that might be result in harmful health effects on pediatric patients. In most nuclear medicine procedures, the amounts of radiation (dose) applied on pediatric patients are limited to certain low levels, but they are contradictory to the mechanistic biologic observations. It had been difficult for most physicians to effectively assess the magnitude of exposure or potential risk due to implementation of nuclear radiations on pediatric treatments. The challenge job is how to make a trade-off between the nuclear radiation dose applied on the pediatric patients and the quality of the diagnostic results, and to select or determine an optimal or minimized effective dose and critical organs to reduce the risk of nuclear radiations [3]. Effective dose provides an approximate indicator of potential detriment from nuclear radiation and should be used as one parameter in evaluating the appropriateness of examinations involving nuclear radiation. In addition, the organ receiving the highest dose is referred to as the critical organ. In fact, effective dose is a calculated quantity and cannot be measured. Multiplying the average organ equivalent dose by the International Commission on Radiological Protection (ICRP) tissue-weighting factor and summing the results over the whole body yields the effective dose [4]. Although effective dose is an average evaluation value, it is still an important parameter in estimation of average potential risks of nuclear radiation on patients.

Because of the popular applications of nuclear medicines on pediatric diagnostics and treatments, remarkable increase in the use of nuclear medical procedures have been shown in the US in recent years [5]. Different techniques and models have been reported and developed to optimize the nuclear radiation dose to reduce the risk of nuclear radiations on patients in last decades [6-8, 12, 14-15]. One of the most important reasons for these developments is to reduce the potential risk of cancers that results from the nuclear radiations exposed from the usage of the nuclear medicine procedures [16-32].

R. Accorsi et al. [9] provided a method to improve the dose regimen in pediatric Positron Emission Tomography

ISBN: 978-1-61208-339-1

(PET). Some other research organizations reported different radiation sources used in nuclear medicines in recent years [10][11]. R. Fazel et al. [13] developed a procedure to use low-dose ionizing radiation in medical image process. Frederic H. Fahey, S. Ted Treves, and S. James Adelstein provided a survey to review most recent developments in using minimized dose to reduce the risk of inducing cancer [16]. R. Loevinger et al. [17] reported a method to calculate the absorbed dose to limit the effects of radiations. Stabin MG and Siegel JA discussed some popular physical models and dose factors for use in internal dose assessment [18]. V.L. Ward et al. developed a method to reduce the effective dose for the pediatric radiation exposure [22]. R.J. Preston reported a linear non-threshold dose-response model and implications for diagnostic radiology procedures [23]. M.J. Gelfand developed a method to reduce the dose applied in pediatric hybrid and planar imaging process [25]. E. Hsaio et al. reported a technique to reduce the radiation dose in Mercaptoacetyltrig (MAG3) renography by enhanced planar processing [27]. Other researchers reported different techniques and methods to reduce radiation exposures in nuclear medicine and medicine image processing [28-32].

However, most of these technologies either utilized simple shapes to present the organs or used more realistic models to estimate the nuclear dose applied on pediatric patients. The former are too simple to provide accurate estimation results, and the latter are too complicated to intensively involve complex calculations. Also, these estimations are averages over a wide range of patients at each age and they are not related to individual differences in anatomy and physiology from the standard models. Application of these pediatric models is problematic because children can vary greatly in body size and habitus. A good model should deal with both the children's age and the body-size to determine the optimal effective dose.

The advantage of using our model as discussed in this paper is that the physicians can easily and quickly calculate and select the optimal or minimized effective dose based on the given age and body-size of the pediatric patient to significantly reduce the effects of nuclear radiations on patients. This kind of model will be more suitable and appropriate for pediatric examination and diagnoses.

The organization of this paper is straightforward. Following the Introduction, the methodology and materials used in this paper are presented. The Fuzzy Inference System (FIS) is introduced in Section 3. In Section 4, the results and conclusion are provided.

II. Materials and Methods

We used the FIS to build a dynamic model to set a mapping relationship between each age, weight and the desired optimal effective dose and critical organs for pediatric patients' groups. All related data and operational parameters used for this model are based on data provided by Fahey et al. [16]. The estimates of critical organ and effective dose for common pediatric nuclear medicine procedures developed in [16] are shown in Table 1; it shows estimated relationships between the pediatric patients' ages, weights and effective doses as well as critical organs for 99mTc-ECD.

TABLE 1. CRITICAL ORGAN AND EFFECTIVE DOSE.

	Max admin act (MBq)	1-y-old	5-y-old	10-y-old	15-y-old	Adult
Mass (kg)		9.7	19.8	33.2	56.8	70
^{99m}Tc-MDP*	740					
Bone surface (mGy)		54.5	46.0	45.6	49.2	46.6
Effective dose (mSv)		2.8	2.9	3.9	4.2	4.2
^{99m}Tc-ECD‡	740					
Bladder wall (mGy)		13.4	23.0	30.5	37.2	37.0
Effective dose (mSv)		4.1	4.6	5.3	5.9	5.7
^{99m}Tc-sestamibi*	740					
Gallbladder (mGy)		32.9	20.9	20.4	27.0	28.9
Effective dose (mSv)		5.4	5.9	6.3	7.2	6.7
^{99m}Tc-MAG3*	370					
Bladder wall (mGy)		17.2	19.8	31.3	44.1	42.7
Effective dose (mSv)		1.2	1.3	2.2	2.8	2.7
^{123}I-MIBG*	370					
Liver (mGy)		16.6	18.5	22.4	25.6	24.8
Effective dose (mSv)		3.4	3.8	4.5	5.0	4.8
^{18}F-FDG‡	370					
Bladder wall (mGy)		25.6	35.9	44.4	48.8	50.5
Effective dose (mSv)		5.2	5.9	6.6	7.3	7.4

* Based on ICRP 80 (25), † Based on ICRP 106 (26).

Max admin act = maximum administered activity is that administered to adult or large child (70 kg) (administered activities for smaller children are scaled by body weight); ECD = ethyl cysteinate dimer; MIBG = meta iodo benzyl guanidine.

It can be seen from Table 1 that it only provided limited information between certain children ages with selected weights and the minimized nuclear effective dose and critical organs. In other words, the relationship or mapping between the children ages, weights and the optimal effective dose and critical organs is incomplete or discrete because it does not provide all optimal effective doses and critical organs for any given children age and weight group.

To improve that incomplete and discrete model, in this study, we will use a FIS to build a complete and continuous model to provide all related optimal effective doses and critical organs for different given children ages and weights groups in a simple and easy way. In fact, we will use the FIS to interpolate the optimal effective dose and critical organs based on the specified age and weight of each child group to simplify the calculation and estimation process for the effective dose and the critical organ.

To make our study simple, we only use the bladder wall with 99mTc-ECD as an example to illustrate how to use FIS to simplify this effective dose calculation and critical organ estimation process. This study can be easily extended to cover all other organs and methods shown in Table 1.

The basic idea behind this model development is based on the fact, that the optimal effective dose and critical organs are not continuous functions for all different given ages and weights located between known ages and weights. Also the relationship between the minimized effective dose, critical organs and different age-weight is ambiguous, or at least it is not a linear function. Therefore, we need to use the fuzzy inference algorithm to derive those optimal effective doses and critical organs for all those 'missed' age-weight pairs. In fact, we use fuzzy inference method to interpolate those optimal effective doses and critical organs for any specified age-weight pair.

III. Fuzzy Inference System

We use given age and weight of the pediatric patient as inputs, and the optimal effective doses and critical organs as outputs for a fuzzy inference system. Therefore, this is a multi-input and multi-output system. Both inputs and output are connected and controlled by the fuzzy system control rules. Fig. 1 shows the block diagram of this fuzzy inference system.

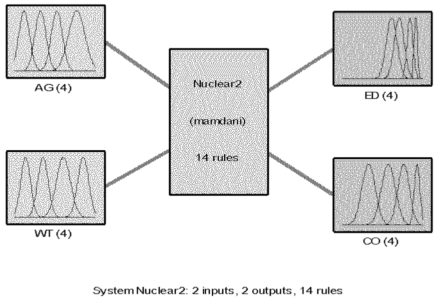


Fig. 1. Block diagram of the fuzzy inference system (FIS).

As for the membership functions for two inputs, pediatric patient Age and Weight, we utilized gaussform as the shape for both of them. Similarly, this shape is also used for two outputs, the optimal effective dose and the critical organ.

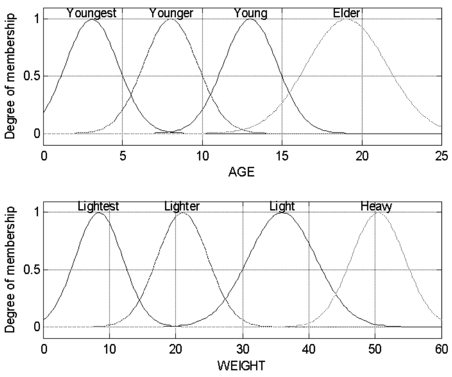


Fig. 2. MFs for two inputs - patient age (AG) and weight (WT).

The membership functions for both inputs (patient's age - AG and weight - WT) are shown in Fig. 2. The membership functions for two outputs (effective dose - ED and critical organ - CO) are shown in Fig. 3, respectively. Those membership functions are derived based on the data provided in [16] for common pediatric nuclear medicine procedures.

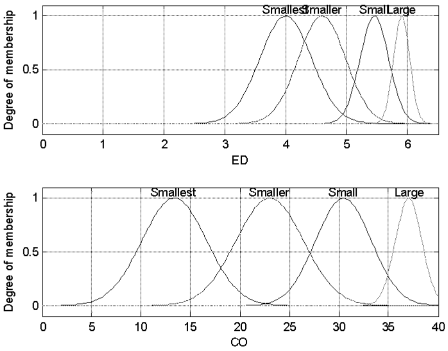


Fig. 3. MFs for outputs - effective dose (ED) and critical organ (CO).

TABLE 2. MF FOR THE PEDIATRIC PATIENT'S AGE

AG (years old)	0 ~ 7	4 ~ 12	9 ~ 17	13 ~ 25
MF	Youngest	Younger	Young	Elder

TABLE 3. MF FOR THE PEDIATRIC PATIENT'S WEIGHT

WT (kg)	0 ~ 17	12 ~ 30	24 ~ 48	40 ~ 60
MF	Lightest	Lighter	Light	Heavy

TABLE 4. MF FOR THE EFFECTIVE DOSE

ED (mSv)	3.0 ~ 5.0	3.7 ~ 5.5	4.9 ~ 6.0	5.6 ~ 6.2
MF	Smallest	Smaller	Small	Large

TABLE 5. MF FOR THE CRITICAL ORGAN

CO (mGy)	5.6 ~ 21.0	15.0 ~ 31.0	23.8 ~ 37.0	33.9 ~ 40.0
MF	Smallest	Smaller	Small	Large

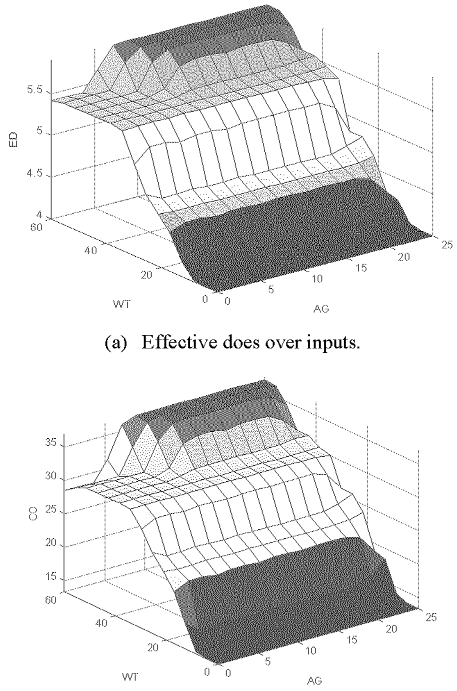
The definitions for the membership functions of the pediatric patient's age and weight are shown in Tables 2 and 3, and the membership functions for effective dose and critical organs are shown in Tables 4 and 5.

For this implementation, fourteen control rules are developed based on the input-output membership functions. These fourteen control rules are shown in Table 6. The surface relationship between the output effective dose (ED) and the inputs, age (AG) and weight (WT), is shown in Fig. 4a, and the surface of the critical organ (CO) over the age (AG) and the weight (WT) is shown in Fig. 4b. The weights of the rules are calculated based on [16].

ISBN: 978-1-61208-339-1

TABLE 6. FOURTEEN FUZZY CONTROL RULES

1. If (AG is Youngest) & (WT is Lightest) then
   (ED is Smallest) & (CO is Smallest) (1)
2. If (AG is Youngest) & (WT is Lighter) then
   (ED is Smaller) & (CO is Smaller) (1)
3. If (AG is Youngest) & (WT is Light) then
   (ED is Small) & (CO is Small) (1)
4. If (AG is Younger) & (WT is Lightest) then
   (ED is Smallest) & (CO is Smallest) (1)
5. If (AG is Younger) & (WT is Lighter) then
   (ED is Smaller) & (CO is Smaller) (1)
6. If (AG is Younger) & (WT is Light) then
   (ED is Small) & (CO is Small) (1)
7. If (AG is Young) & (WT is Lightest) then
   (ED is Smallest) & (CO is Smallest) (1)
8. If (AG is Young) & (WT is Lighter) then
   (ED is Smaller) & (CO is Smaller) (1)
9. If (AG is Young) & (WT is Light) then
   (ED is Small) & (CO is Small) (1)
10. If (AG is Young) & (WT is Heavy) then
   (ED is Large) & (CO is Large) (1)
11. If (AG is Elder) & (WT is Lightest) then
   (ED is Smallest) & (CO is Smallest) (1)
12. If (AG is Elder) & (WT is Lighter) then
   (ED is Smaller) & (CO is Smaller) (1)
13. If (AG is Elder) & (WT is Light) then
   (ED is Small) & (CO is Small) (1)
14. If (AG is Elder) & (WT is Heavy) then
   (ED is Large) & (CO is Large) (1)



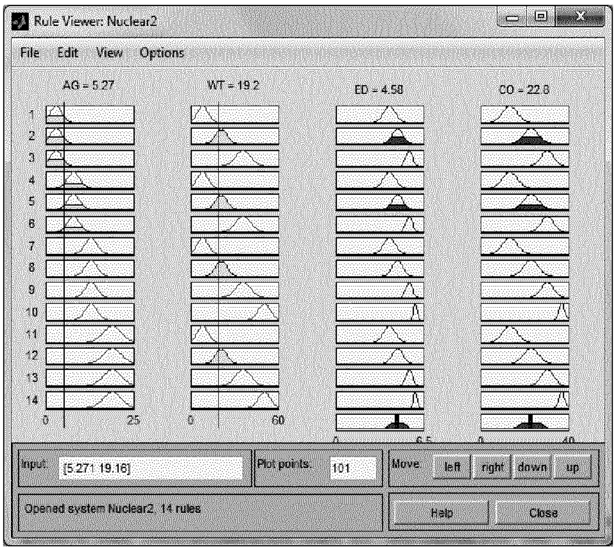
(a) Effective does over inputs.

(b) Critical Organ over inputs.

Fig. 4. The surfaces between inputs and outputs.

IV. Results and Conclusion

Based on the membership functions of two inputs, patient's age and weight, and the membership functions of two outputs, effective dose and critical organ, the desired optimal effective dose and the estimated critical organ for the given patient's age and weight can be easily determined and obtained directly from the fuzzy rule relationship. Fig. 5 shows this kind of model for the calculation of optimal effective dose and critical organs used in pediatric bladder wall inspection using the nuclear medicine procedures.



Input: [5.271 19.16]

Plot points: 101

Opened system Nuclear2, 14 rules

Fig. 5. The fuzzy rule mapping between the inputs and the outputs.

In Fig. 5, a typical pediatric patient age (5.27 years old) and weight (19.2 kg) are selected. The related optimal effective dose (4.58 mSv) and the critical organ (22.8 mGy) are determined directly from this fuzzy rule relationship.

During the implementation process, the vertical bars on both inputs, patient's age and weight, can be moved by the pediatric physician to either left or right to select the specified age and weight group of pediatric patients, and the desired optimal effective dose and critical organ can be easily determined directly from this fuzzy input-output rules relationship map. This model provides great flexibility and simplicity to determine the optimal effective dose and critical organs for common pediatric nuclear medicine procedures.

We can also easily build a similar FIS model by using the data provided in [16] and the method proposed by Ying et al. [33] to determine the related optimal effective doses and the desired critical organ for all other kinds of pediatric organs' nuclear medicine procedures.

A flexible and simple model used to set a fuzzy mapping relationship between the pediatric patients' age-weight and the optimal effective dose and critical organs is developed in this study to enable pediatric physicians to easily and directly determine the optimal effective doses and critical organs for the common pediatric nuclear medicine procedures. Compared with some other traditional methods, the advantage of using this model is that the pediatric physicians can easily and directly obtain the desired minimized effective dose and the critical organs from the fuzzy rule relationship based on the given group of pediatric patients' data, such as ages and weights.

ACKNOWLEDGMENT

Special thanks should be given to F. H. Fahey, S. Ted Treves, and S. James Adelstein for their permission to allow us to use their table, Table 1, in one of their papers, Minimizing and Communicating Radiation Risk in Pediatric Nuclear Medicine published in Journal of Nuclear Medicine in March 1, 2012.

REFERENCES

[1] S.T. Treves, Pediatric Nuclear Medicine. New York, NY, Springer, 2007.

[2] S. T. Treves, A. Baker, F.H. Fahey, X. Cao, R. T. Davis, L. A. Drubach, F. D. Grant and K. Zukotynski, Nuclear medicine in the first year of life. Journal of Nuclear Medicine. 2011, 52, pp. 905–925.

[3] Committee to Assess Health Risks from Exposure to Low Levels of Ionizing Radiation, National Research Council. Health Risks from Exposure to Low Levels of Ionizing Radiation, BEIR VII Phase 2. Washington, DC, National Research Council of the National Academies, 2006.

[4] A. Fred, Mettler Jr, W. Huda, T. Yoshizumi, and M. Mahesh, "Effective Doses in Radiology and Diagnostic Nuclear Medicine: A Catalog", July 2008, Radiology, 248, pp. 254-263.

[5] National Council on Radiation Protection and Measurement. Ionizing Radiation Exposure of the Population of the United States: Report NCRP 160. Washington, DC, National Council on Radiation Protection and Measurement, 2009.

[6] M. C. Eckerman, Specific Absorbed Fractions of Energy at Various Ages. Oak Ridge, TN, Oak Ridge National Laboratories, 1987, ORNL/TM-8381.

[7] R. Nakazato, et al., "Myocardial Perfusion Imaging with a Solid-State Camera: Simulation of a Very Low Dose Imaging Protocol", J. of Nuclear Medicine, March 1, 2013 vol. 54, no. 3, pp. 373-379.

[8] X. Setoain, et al, "Validation of an Automatic Dose Injection System for Ictal SPECT in Epilepsy", J. of Nuclear Medicine, February 1, 2012 vol. 53, no. 2, pp. 324-329.

[9] R. Accorsi, J. S. Karp, and S. Surti, "Improved Dose Regimen in Pediatric PET", J. of Nuclear Medicine, February 2010, vol. 51, no. 2, pp. 293-300.

[10] Sources and Effects of Ionizing Radiation: UNSCEAR 2008 Report. Volume I: Sources—Report to the General Assembly Scientific Annexes A, B. New York, NY: United Nations; 2010.

[11] F.A. Mettler et al., Radiologic and nuclear medicine studies in the United States and worldwide, frequency, radiation dose, and comparison with other radiation sources, 1950-2007, *Radiology.* 2009, 253, pp. 520-531.

[12] A.L Dorfman et al., "Use of medical imaging procedures with ionizing radiation in children: a population-based study", Arch Pediatr Adolesc Medicine. 2011, 165, pp. 458-464.

[13] R. Fazel et al., "Exposure to low-dose ionizing radiation from medical imaging procedure", N Engl J Med. 2009;361: pp. 849–857.

[14] L. Kowalczyk, Is all that scanning putting us at risk? Boston Globe. September 14, 2009, G6.

[15] E.S. Amis and P.F. Butler., ACR white paper on radiation dose in medicine: three years later. J. of Am Coll Radiol. 2010, 7, pp. 865–870.

[16] F. H. Fahey, S. Ted Treves, and S. James Adelstein, "Minimizing and Communicating Radiation Risk in Pediatric Nuclear Medicine", Journal of Nuclear Medicine, March 1, 2012, vol. 40, no. 1, pp. 13-24.

[17] R. Loevinger and T.F. Budinger, MIRD Primer for Absorbed Dose Calculations (Revised Edition) Reston, VA, Society of Nuclear Medicine, 1991.

[18] M.G. Stabin and J.A. Siegel., "Physical models and dose factors for use in internal dose assessment", Health Phys. 2003, 85, pp. 294-310.

[19] G. Xu and K.F. Eckerman eds., Handbook of Anatomical Models for Radiation Dosimetry. Boca Raton, FL, *CRC Press,* 2009.

[20] S. Whalen, C. Lee, J. Williams, and W.E. Bolch, Anthropomorphic approaches and their uncertainties to assigning computational phantoms to individual patients in pediatric dosimetry studies, Phys Med Biol. 2008, 53,pp. 453-471.

[21] M.G. Stabin, Internal Dosimetry in Pediatric Nuclear Medicine. 3rd ed., New York, NY, Springer, 2007, pp. 513-520.

[22] V.L. Ward et al., "Pediatric radiation exposure and effective dose reduction during voiding cystourethrography", Radiology, 2008, 249, pp. 1002-1009.

[23] R.J. Preston, "Update on linear non-threshold dose-response model and implications for diagnostic radiology procedures", Health Phys., 2008, 95, pp. 541-546.

[24] K.E. Thomas et al., "Assessment of radiation dose awareness among pediatricians", Pediatr Radiol., 2006, 36, pp. 823-832.

[25] M.J. Gelfand, "Dose reduction in pediatric hybrid and planar imaging", Q J. of Nucl Med Mol Imaging, 2010, 54, pp. 379-388.

[26] S.T. Treves, R.T. Davis, and F.H. Fahey, "Administered radiopharmaceutical doses in children: a survey of 13 pediatric hospitals in North America", J. Nucl Med., 2008, 49, pp. 1024-1027.

[27] E. Hsaio, et al., "Reduction in radiation dose in MAG3 renography by enhanced planar processing", Radiology, December 2011, 261, pp. 907-915.

[28] M.J. Gelfand, M.T. Parisi, and S.T. Treves, "Pediatric radiopharmaceutical administered doses: 2010 North American consensus guidelines", J. of Nucl Med., 2011, 52, pp. 318-322.

[29] "Dose Guidelines for Pediatric Nuclear Medicine", http://www.asrt.org/main/news-research/press-room/2010/10/14/DoseGuidelinesforPediatricNuclearMedicine, Oct. 2010.

[30] G. R Small, Benjamin JW Chow, and Terrence D Ruddy, "Low-dose Cardiac Imaging", Expert Rev Cardiovasc Ther., 2012, 10(1), pp. 89-104.

[31] "Reducing Radiation Exposure in Nuclear Medicine by Novel Processing Techniques", http://www.medscape.com/viewarticle/755808_25, 2012.

[32] H. Hricak, et al., "Managing radiation use in medical imaging: a multifaceted challenge", Radiology, 2011, 258, pp. 889-905.

[33] B. Ying, D. Wang, and S. Gupta. "Estimate the Minimized Effective Dose and Critical Organ in Pediatric Nuclear Medicine." American Journal of Medical Case Reports vol 2., no.1 (2014): pp. 4-9.

Level-Synchronous Parallel Breadth-First Search Algorithms For Multicore and Multiprocessor Systems

Rudolf Berrendorf and Matthias Makulla
Computer Science Department
Bonn-Rhein-Sieg University
Sankt Augustin, Germany
e-mail: rudolf.berrendorf@h-brs.de, mathias.makulla@h-brs.de

***Abstract*—Breadth-First Search (BFS) is a graph traversal technique used in many applications as a building block, e.g., to systematically explore a search space. For modern multicore processors and as application graphs get larger, well-performing parallel algorithms are favourable. In this paper, we systematically evaluate an important class of parallel BFS algorithms and discuss programming optimization techniques for their implementation. We concentrate our discussion on level-synchronous algorithms for larger multicore and multiprocessor systems. In our results, we show that for small core counts many of these algorithms show rather similar behaviour. But, for large core counts and large graphs, there are considerable differences in performance and scalability influenced by several factors. This paper gives advice, which algorithm should be used under which circumstances.**

***Index Terms*—parallel breadth-first search; BFS; NUMA; memory bandwidth; data locality**

I. Introduction

BFS is a visiting strategy for all vertices of a graph. BFS is most often used as a building block for many other graph algorithms, including shortest paths, connected components, bipartite graphs, maximum flow, and others [1]. Additionally, BFS is used in many application areas where certain application aspects are modelled by a graph that needs to be traversed according to the BFS visiting pattern. Amongst others, exploring state space in model checking, image processing, investigations of social and semantic graphs, machine learning are such application areas [2].

Many parallel BFS algorithms got published (see Section III for a comprehensive overview including references), all with certain scenarios in mind, e.g., large distributed memory systems with the message passing programming model [3], GPU's (Graphic Processing Unit) with a different parallel programming model [4], or randomized algorithms for fast, but possibly sub-optimal results [5]. Such original work often contains performance data for the newly published algorithm on a certain system, but often just for the new approach, or taking only some parameters in the design space into account [6] [7]. To the best of our knowledge, there is no rigid comparison that systematically evaluates relevant parallel BFS algorithms in detail in the design space with respect to parameters that may influence the performance and/or scalability and give advice which algorithm is best suited for which application scenario. In this paper, BFS algorithms of a class with a large practical impact (level-synchronous algorithms for shared memory parallel systems) are systematically compared to each other.

The paper first gives an overview on parallel BFS algorithms and classifies them. Second, and this is the main contribution of the paper, a selection of level-synchronous algorithms relevant for the important class of multicore and multiprocessors systems with shared memory are systematically evaluated with respect to performance and scalability. The results show that there are significant differences between algorithms for certain constellations, mainly influenced by graph properties and the number of processors / cores used. No single algorithm performs best in all situations. We give advice under which circumstances which algorithms are favourable.

The paper is structured as follows. First, a BFS problem definition is given. Section III gives a comprehensive overview on parallel BFS algorithms with an emphasis on level synchronous algorithms for shared memory systems. Section IV prescribes algorithms in detail that are of concern in this paper. Section V describes our experimental setup, and, in section VI, the evaluation results are discussed, followed by a conclusion.

II. Breadth-First Search Graph Traversal

We are interested in undirected graphs $G = (V,E)$, where $V = \{v_1,...,v_n\}$ is a set of vertices and $E = \{e_1,...,e_m\}$ is a set of edges. An edge e is given by an unordered pair $e = (v_i, v_j)$ with $v_i, v_j \in V$. The number of vertices of a graph will be denoted by $|V| = n$ and the number of edges is $|E| = m$.

Assume a connected graph and a source vertex $v_0 \in V$. For each vertex $u \in V$ define $depth(u)$ as the number of edges on the shortest path from v_0 to u, i.e., the edge distance from v_0. With $depth(G)$ we denote the depth of a graph G defined as the maximum depth of any vertex in the graph *relative to the given source vertex*. Please be aware that this may be different to the diameter of a graph, the largest distance between *any* two vertices.

The problem of BFS for a given graph $G = (V,E)$ and a source vertex $v_0 \in V$ is to visit each vertex in a way such that a vertex v_1 must be visited before any vertex v_2 with $depth(v_1) < depth(v_2)$. As a result of a BFS traversal, either the level of each vertex is determined or a (non-unique) BFS spanning tree with a father-linkage of each vertex is created. Both variants can be handled by BFS algorithms with small modifications and without extra computational effort. The problem can be easily extended and handled with directed or unconnected graphs. A sequential solution to the problem can be found in textbooks based on a queue where all non-visited adjacent vertices of a visited vertex are enqueued [1]. The computational complexity is $O(|V|+|E|)$.

III. Parallel BFS Algorithms and Related Work

We combine in our BFS implementations presented later in Section IV several existing algorithmic approaches and optimization techniques. Therefore, the presentation of related work has to be intermingled with an overview on parallel BFS algorithms itself.

In the design of a parallel BFS algorithm, different challenges might be encountered. As the computational density for BFS is rather low, BFS is memory bandwidth limited for large graphs and therefore bandwidth has to be handled with care. Additionally, memory accesses and work distribution are both irregluar and data-dependent. Therefore, in large NUMA systems (Non-Uniform Memory Access [8]) data layout and memory access should respect processor locality. In multicore multiprocessor systems, things get even more complicated, as several cores share higher level caches and NUMA-node memory, but have distinct and private lower-level caches.

A more general problem for many parallel algorithms including BFS is a sufficient load balance when static partitioning is not sufficient. Even when an appropriate mechanism for load balancing is deployed, graphs might only supply a limited amount of parallelism. This aspect especially affects the popular level-synchronous approaches for parallel BFS we concentrate on later.

In BFS algorithms, housekeeping has to be done on visited / unvisited vertices with several possibilities how to do that. A rough classification of algorithms can be achieved by looking at these strategies. Some of them are based on special container structures where information has to be inserted and deleted. Scalability and administrative overhead of these containers are of interest. Many algorithms can be classified into two groups: *container centric* and *vertex centric* approaches.

A. Container Centric Approaches

The emphasis in this paper is on level-synchronous algorithms where data structures are used, which store the current and the next vertex frontier. Generally speaking, these approaches deploy two identical containers (*current* and *next*) whose roles are swapped at the end of each iteration. Usually, each container is accessed in a concurring manner such that the handling/avoidance of synchronized access becomes crucial. Container centric approaches are eligible for dynamic load balancing but are sensible to data locality on NUMA systems. Container centric approaches for BFS can be found in some parallel graph libraries [9] [10].

For level synchronous approaches, a simple list is a sufficient container. There are approaches, in which each thread manages two private lists to store the vertex frontiers and uses additional lists as buffers for communication [3] [11]. This approach deploys a static one dimensional partitioning of the graph's vertices and therefore supports data locality. But this approach completely neglects load balancing mechanisms. The very reverse would be an algorithm, which focuses on load balancing. This can be achieved by using special lists that allow concurrent access of multiple threads. In contrast to the thread private lists of the previous approach, two global lists are used to store the vertex frontiers. The threads then concurrently work on these lists and implicit load balancing can be achieved. Concurrent lock-free lists can be efficiently implemented with an atomic compare-and-swap operation.

It is possible to combine both previous approaches and create a well optimized method for NUMA architectures [6] [7] (this paper came too late to our knowledge to include it in our evaluation). Furthermore, lists can be utilised to implement container centric approaches on special hardware platforms as graphic accelerators with warp centric programming [4].

Besides strict FIFO (First-In-First-Out) and relaxed list data structures, other specialized containers may be used. A notable example is the *bag* data structure [12], which is optimized for a recursive, task parallel formulation of a parallel BFS algorithm. This data structure allows an elegant, object-oriented implementation with implicit dynamic load balancing, but which regrettably lacks data locality.

B. Vertex Centric Approaches

A vertex centric approach achieves parallelism by assigning a parallel entity (e.g., a thread) to each vertex of the graph. Subsequently, an algorithm repeatedly iterates over all vertices of the graph. As each vertex is mapped to a parallel entity, this iteration can be parallelised. When processing a vertex, its neighbours are inspected and if unvisited, marked as part of the next vertex frontier. The worst case complexity for this approach is $O(n^2)$ for degenerated graphs (e.g., linear lists). This vertex centric approach might work well only, if the graph depth is very low.

A vertex centric approach does not need any additional data structure beside the graph itself and the resulting $level/father$-array that is often used to keep track of visited vertices. Besides barrier synchronisation at the end of a level iteration, a vertex centric approach does with some care not need any additional synchronisation. The implementation is therefore rather simple and straightforward. The disadvantages of vertex centric approaches are the lacking mechanisms for load balancing and graphs with large depth (e.g., a linear list).

But this overall approach makes it well-suited for GPU's where each thread is mapped to exactly one vertex [13] [14]. This approach can be optimized further by using hierarchical vertex frontiers to utilize the memory hierarchy of a graphic accelerator, and by using hierarchical thread alignment to reduce the overhead caused by frequent kernel restarts [15].

Their linear memory access and the possibility to take care of data locality allow vertex centric approaches to be efficiently implemented on NUMA machines [16]. Combined with a proper partitioning, they are also suitable for distributed systems, as the overhead in communication is rather low.

C. Other Approaches

The discussion in this paper concentrates on level-synchronous parallel BFS algorithms for shared-memory parallelism. There are parallel algorithms published that use different approaches or that are designed for other parallel architectures in mind. In [5], a probabilistic algorithm is shown that finds a BFS tree with high probability and that works in practice well even with high-diameter graphs. Beamer et.al. [17] combines a level-synchronous top-down approach with a vertex-oriented bottom-up approach where a heuristic switches between the two alternatives; this algorithm shows for small world graphs very good performance. Yasui et.al. [18] explores this approach in more detail for multicore systems. In [19], a fast GPU algorithm is introduced that combines fast primive operations like prefix sums available with highly-optimized libraries. A task-based approach for a combination of CPU/ GPU is presented in Munguia et.al. [20].

For distributed memory systems, the partitioning of the graph is crucial. Basically, the two main strategies are one dimensional partitioning of the vertices and two dimensional edge partitioning [3]. The first approach is suited for small distributed and most shared memory systems, while the second one is viable for large distributed systems. Optimizations of these approaches combine threads and processes in a hybrid environment [21] and use asynchronous communication [22] to tolerate communication latencies.

D. Common extensions and optimizations

An optimization applicable to some algorithms is the use of a bitmap to keep track of visited vertices in a previous iteration [6]. The intention is to keep more information on visited vertices in a higher level of the cache hierarchy.

Fine-grained tuning like memory prefetching can be used to tackle latency problems [7] (but which might produce even more pressure on memory bandwidth).

Besides implicit load balancing of some container centric approaches, there exist additional methods. One is based on a logical ring topology [23] of the involved threads. Each thread keeps track of its neighbour's workload and supplies it with additional work, if it should be idle. Another approach to adapt the algorithm to the topology of the graph monitors the size of the next vertex frontier. At the end of an iteration, the number of active threads is adjusted to match the workload of the coming iteration [11].

IV. Evaluated Algorithms

In our evaluation, we used the following parallel algorithms, each representing certain points in the described algorithm design space for shared memory systems, with an emphasis on level-synchronous algorithms:

- `global`: vertex-centric strategy as described in Section III-B, with parallel iterations over *all vertices on each level* [16]. As pointed out already, this will only work for graphs with a very low depth.
- `graph500`: OpenMP reference implementation in the Graph500 benchmark [9] using a single array list with atomic Compare-And-Swap (CAS) and Fetch-And-Add accesses to insert chunks of vertices. Vertex insertion into core-local chunks is done without synchronized accesses. Only the insertion of a full chunk into the global list has to be done in a synchronized manner. All vertices of a full chunk get copied to the global array list.
- `bag`: using OpenMP [24] tasks and two bag containers as described in [12]. This approach implicitly deploys load balancing mechanisms. Additionally, we implemented a Cilk++ version as in the the original paper that didn't perform better than the OpenMP version.
- `list`: deploys two chunked linear lists with thread safe manipulators based on CAS operations. Threads concurrently remove chunks from the current node frontier and insert unvisited vertices into private chunks. Once a chunk is full, it is inserted into the next node frontier, relaxing concurrent access. The main difference to `graph500` ist that vertices are not copied to a global list but rather a whole chunk gets inserted (updating pointers only). There is some additional overhead, if local chunks get filled only partially.
- `socketlist`: extends the previous approach to respect data locality and NUMA awareness. The data is logically and physically distributed to all NUMA-nodes (i.e., CPU sockets). Each thread primarily processes vertices from its own NUMA-node list where the lists from the previous approach are used for equal distribution of work. If a NUMA-node runs out of work, work is stolen from overloaded NUMA-nodes [6].
- `bitmap`: further refinement and combination of the previous two approaches. A bitmap is used to keep track of visited vertices to reduce memory bandwidth. Again, built-in atomic CAS operations are used to synchronize concurrent access [6].

The first algorithm is vertex-centric, all others are level-synchronous container-centric in our classification and utilize parallelism over the current vertex front. The last three implementations use a programming technique to trade (slightly more) redundant work against atomic operations as described in [25]. `socketlist` is the first in the algorithm list that pays attention to the NUMA memory hierarchy, `bitmap` additionally tries to reduce memory bandwidth by using an additional bitmap to keep track of the binary information whether a vertex is visited or not.

V. Experimental Setup

In this section, we specify our parallel system test environment, describe classes of graphs and chosen graph representatives in this classes.

A. Test Environment

We used in our tests different systems, the largest one a 64-way AMD-6272 Interlagos based system with 128 GB shared memory organised in 4 NUMA nodes, each with 16 cores (1.9 GHz). Two other systems are Intel based with 2 NUMA nodes each (Intel-IB: 48-way E5-2697 Ivy bridge EP at 2.7 GHz, Intel-SB: 32-way E5-2670 Sandy-Bridge EP at 2.6 GHz). We will focus our discussion on the larger Interlagos system and discuss in section VI-C the influence of the system details.

B. Graphs

It is obvious that graph topology will have a significant influence on the performance of parallel BFS algorithms. We used beside some real graphs synthetically generated pseudo-random graphs that guarantee certain topological properties. R-MAT [26] is such a graph generator with parameters a, b, c influencing the topology and clustering properties of the generated graph (see [26] for details). R-MAT graphs are mostly used to model scale-free graphs. We used in our tests graphs of the following classes:

- Graphs with a very low average and maximum vertex degree resulting in a rather high graph depth and limited vertex fronts. A representative for this class is the road network `road-europe`.
- Graphs with a moderate average and maximum vertex degree. For this class we used Delaunay graphs representing Delaunay triangulations of random points (`delaynay`) and a graph for a 3D PDE-constraint optimization problem (`nlpkkt240`).
- Graphs with a large variation of degrees including few very large vertex degrees. Related to the graph size, they have a smaller graph depth. For this class of graphs we used a real social network (`friendster`), link information for web pages (`wikipedia`), and synthetically generated Kronecker R-MAT graphs with different vertex and edge counts and three R-MAT parameter sets. The first parameter set named 30 is $a = 0.3, b = 0.25, c = 0.25$, the second parameter set 45 is $a = 0.45, b = 0.25, c = 0.15$, and the third parameter set 57 is $a = 0.57, b = 0.19, c = 0.19$. The default for all our RMAT-graphs is the parameter set 57; the graphs with the suffix `-30` and `-45` are generated with the corresponding parameter sets.

All our test graphs are connected, for R-MAT graphs guaranteed with $n-1$ artificial edges connecting vertex i with vertex $i+1$. Some important graph properties for the graphs used are given in table I. For a general discussion on degree distributions of R-MAT graphs see [27].

VI. Results

In this section, we discuss our results for the described test environment. Performance results will be given in *Million Traversed Edges Per Second* $MTEPS := m/t/10^6$, where m is the number of edges and t is the time an algorithm takes. MTEPS is a common metric for BFS performance [9] (higher is better). In an undirected graph representing an edge internally with two edges (u,v) and (v,u) only half of the internal edges are counted in this metric.

In the following discussion on results, we distinguish between different views on the problem. It is not possible to show all our results in this paper in detail (3 parallel systems, 35 different graphs, up to 11 thread counts, 32/64 bit versions, different compilers / compiler switches). Rather than that, we summarize results and show only interesting or representative aspects in detail.

TABLE I: CHARACTERISTICS FOR SOME OF THE USED GRAPHS.

			degree		graph
graph name	$\lvert V\rvert \times 10^6$	$\lvert E\rvert \times 10^6$	avg.	max.	depth
delaunay (from [28])	16.7	100.6	6	26	1650
nlpkkt240 (from [29])	27.9	802.4	28.6	29	242
road-europe (from [28])	50.9	108.1	2.1	13	17345
wikipedia (from [29])	3.5	45	12.6	7061	459
friendster (from [30])	65.6	3612	55	5214	22
RMAT-1M-10M	1	10	10	43178	400
RMAT-1M-10M-45	1	10	10	4726	16
RMAT-1M-10M-30	1	10	10	107	11
RMAT-1M-100M	1	100	100	530504	91
RMAT-1M-100M-45	1	100	100	58797	8
RMAT-1M-100M-30	1	100	100	1390	9
RMAT-1M-1G	1	1000	1000	5406970	27
RMAT-1M-1G-45	1	1000	1000	599399	8
RMAT-1M-1G-30	1	1000	1000	13959	8
RMAT-100M-1G	100	1000	10	636217	3328
RMAT-100M-2G	100	2000	20	1431295	1932
RMAT-100M-3G	100	3000	30	2227778	1670
RMAT-100M-4G	100	4000	40	3024348	1506

On large and more dense graphs, MTEPS values are generally higher than on very sparse graphs. The MTEPS numbers vary between less than 1 and approx. 3,500, depending on the graph. This is due to the fact that in denser graphs many visited edges do not generate an *additional* entry (and therefore work) in a container of unvisited vertices. This observation is not true for `global`, where in all levels all vertices get traversed.

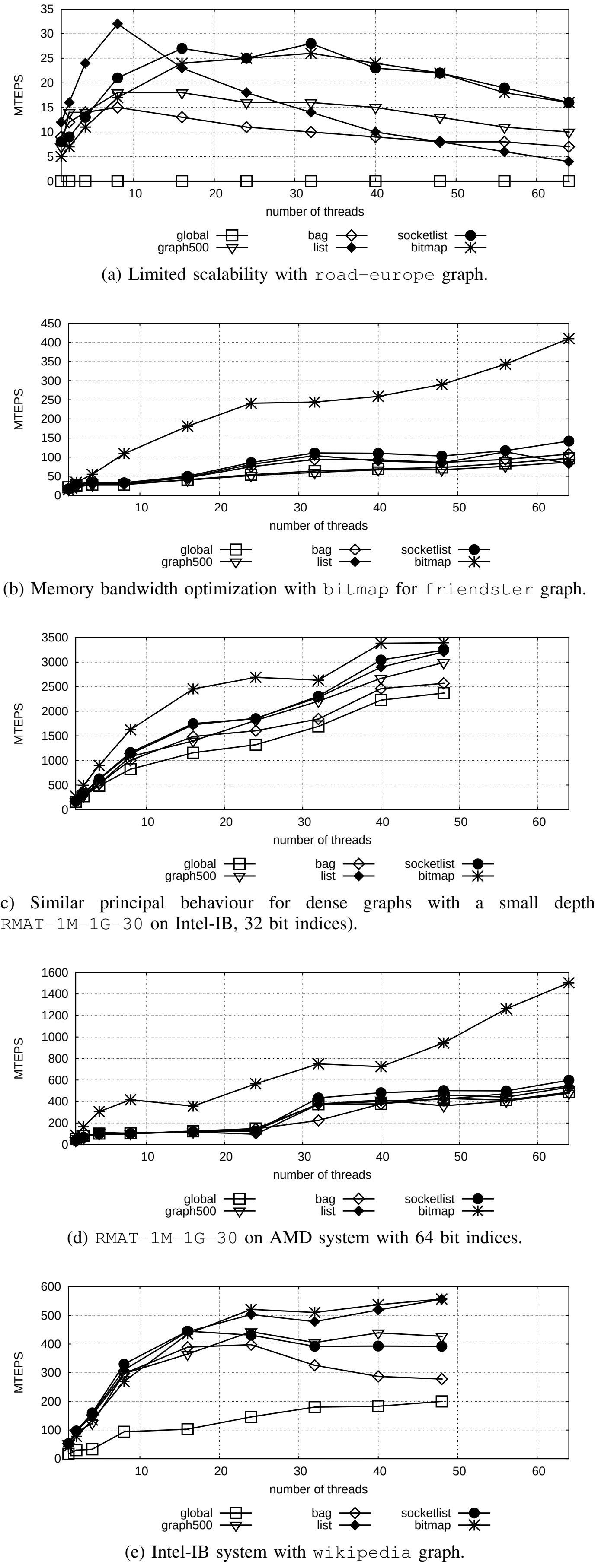
A. Graph Properties and Scalability

In terms of scalability, parallel algorithms need enough parallel work to feed all threads. For graphs with limitating properties, such as small vertex degrees or small total number of vertices / edges, there are problems to feed many parallel threads. Additionally, congestion in accessing smaller shared data structures arise. For such graphs (road network, the delaunay graph and partially small RMAT-graphs), for *all* analysed algorithms performance is limited or even drops as soon the number of threads is beyond some threshold; on *all* of our systems around 8-16 threads. Figure 1a shows the worst case of such an behaviour with `road-europe`. Figure 2 shows different vertex frontier sizes for 3 graphs.

For large graphs and/or high vertex degrees (all larger R-MAT graphs, `friendster`, `nlpkkt240`), the results were quite different from that and all algorithms other than `global` showed on nearly all such graphs and with few exceptions a continuous but in detail different performance increase over all thread counts (see detailled discussion below). Best speedups reach nearly 40 (`bitmap` with `RMAT-1M-1G-30`) on the 64-way parallel system.

B. Algorithms

For small thread counts up to 4-8, all algorithms other than `global` show with few exceptions and within a factor of 2 comparable results in absolute performance and behaviour. But, for large thread counts, algorithm behaviour can be quite



(a) Limited scalability with `road-europe` graph.

(b) Memory bandwidth optimization with `bitmap` for `friendster` graph.

(c) Similar principal behaviour for dense graphs with a small depth (`RMAT-1M-1G-30` on Intel-IB, 32 bit indices).

(d) `RMAT-1M-1G-30` on AMD system with 64 bit indices.

(e) Intel-IB system with `wikipedia` graph.

Fig. 1: Selected performance results.

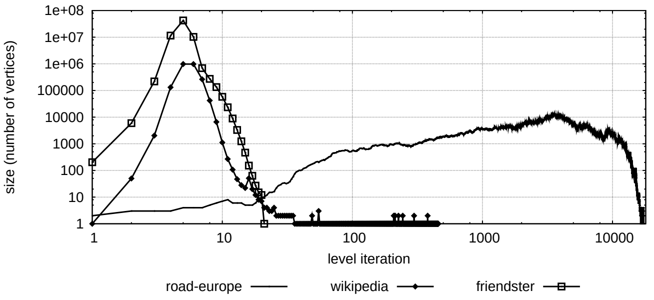


Fig. 2: Dynamic sizes of some vertex frontiers (and potential parallelism).

different. We concentrate, therefore, the following discussions on individual algorithms primarily on large thread counts.

The algorithm `global` has a very different approach than all other algorithms which can be also easily seen in the results. For large graphs with low vertex degrees, this algorithm performs extremely poor as many level-iterations are necessary (e.g., factor 100 slower for road graphs compared to the second worst algorithm; see Figure 1a). The algorithm is only competitive on the systems we used if the graph is very small (no startup overhead with complex data structures) and the graph depth is very low resulting in only a few level-iterations (e.g., less than 10).

The `graph500` algorithm uses atomic operations to increment the position where (a chunk of) vertices get to be inserted into the new vertex front. Additionally, all vertices of a local chunk get copied to the global list (vertex front). This can be fast as long as the number of processors is small. But, as the thread number increases, the cost *per atomic operation* increases [25], and therefore, the performance drops often significantly relative to other algorithms. Additionally, this algorithm does not respect data/NUMA locality on copying vertices which gets a problem with large thread counts.

Algorithm `bag` shows only good results for small thread counts or dense graphs. Similar to `graph500`, this algorithm is not locality / NUMA aware. The bag data structure is based on smaller substructures. Because of the recursive and task parallel nature of the algorithm, the connection between the allocating thread and the data is lost, destroying data locality as the thread count increases. Respecting locality is delegated solely to the run-time system mapping tasks to cores / NUMA nodes. Explicit affinity constructs as in the newest OpenMP version 4.0 [24] could be interesting for that to optimize this algorithm for sparser graphs or many threads.

The simple `list` algorithm has good performance values for small thread counts. But for many threads, `list` performs rather poor on graphs with high vertex degrees. Reasons are implementation specific the use of atomic operations for insert / remove of full/final chunks and that in such vertex lists processor locality is not properly respected. When a thread allocates memory for a vertex chunk and inserts this chunk into the next node frontier, it might be dequeued by another thread in the next level iteration. This thread might be executed on a different NUMA-node, which results in remote memory accesses. This problem becomes larger with increasing thread/processor counts.

The `socketlist` approach improves the list idea with respect to data locality. For small thread counts, this is a small additional overhead, but, for larger thread counts, the advantage is obvious looking at the cache miss and remote access penalty time of current and future processors (see all figures).

The additional overhead of the `bitmap` algorithm makes this algorithm with few threads even somewhat slower than some other algorithms. But the real advantage shows off with very large graphs and large thread counts, where even higher level caches are not sufficient to buffer vertex fronts. The performance difference to all other algorithms can be significant and is even higher with denser graphs (see Figures 1b, 1c, and 1d).

C. *Influence of the system architecture*

As described in Section V, we used in our tests different systems but concentrate our discussions so far on results on the largest AMD system. While the principle system architecture on Intel and AMD systems got in the last years rather similar, implementation details, e.g., on cache coherence, atomic operations and cache sizes are quite different.

While the Intel systems were 2 socket systems, the AMD system was a 4 socket system, and that showed (as expected) more sensibility to locality / NUMA. Hyperthreading on Intel systems gave improvements only for large RMAT graphs. Switching from 64 to 32 bit indices (which restricts the number of addressable edges and vertices in a graph) showed improvements due to lower memory bandwidth requirements. These improvements were around 20-30% for all algorithms other than `bitmap`.

VII. Conclusions

In our evaluation for a selection of parallel level synchronous BFS algorithms for shared memory systems, we showed that for small systems / a limited number of threads all algorithms other than `global` behaved almost always rather similar, including absolute performance.

But using large parallel NUMA-systems with a deep memory hierarchy, the evaluated algorithms show often significant differences. Here, the NUMA-aware algorithms `socketlist` and `bitmap` showed constantly good performance and good scalability, if vertex fronts are large enough. Both algorithms utilise dynamic load balancing combined with locality handling, this combionation is a necessity on larger NUMA systems.

References

[1] R. Sedgewick, Algorithms in C++, Part 5: Graph Algorithms, 3rd ed. Addison-Wesley Professional, 2001.

[2] C. Wilson, B. Boe, A. Sala, K. Puttaswamy, and B. Zhao, “User interactions in social networks and their implications,” in Eurosys, 2009, pp. 205–218.

[3] A. Yoo, E. Chow, K. Henderson, W. McLendon, B. Hendrickson, and U. Catalyurek, “A scalable distributed parallel breadth-first search algorithm on BlueGene/L,” in ACM/IEEE Supercomputing, 2005, pp. 25–44.

[4] S. Hong, S. Kim, T. Oguntebi, and K. Olukotun, “Accelerating CUDA graph algorithms at maximum warp,” in 16th ACM Symp. PPoPP, 2011, pp. 267–276.

[5] J. D. Ullman and M. Yannakakis, “High-probability parallel transitive closure algorithms,” SIAM Journal Computing, vol. 20, no. 1, 1991, pp. 100–125.

[6] V. Agarwal, F. Petrini, D. Pasetto, and D. A. Bader, “Scalable graph exploration on multicore processors,” in ACM/IEEE Intl. Conf. HPCNSA, 2010, pp. 1–11.

[7] J. Chhungani, N. Satish, C. Kim, J. Sewall, and P. Dubey, “Fast and efficient graph traversal algorithm for CPUs: Maximizing single-node efficiency,” in Proc. 26th IEEE IPDPS, 2012, pp. 378–389.

[8] J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, 5th ed. Morgan Kaufmann Publishers, Inc., 2012.

[9] Graph 500 Comitee, Graph 500 Benchmark Suite, http://www.graph500.org/, retrieved: 08.03.2014.

[10] D. Bader and K. Madduri, “SNAP, small-world network analysis and partitioning: an open-source parallel graph framework for the exploration of large-scale networks,” in 22nd IEEE Intl. Symp. on Parallel and Distributed Processing, 2008, pp. 1–12.

[11] Y. Xia and V. Prasanna, “Topologically adaptive parallel breadth-first search on multicore processors,” in 21st Intl. Conf. on Parallel and Distributed Computing and Systems, 2009, pp. 1–8.

[12] C. E. Leiserson and T. B. Schardl, “A work-efficient parallel breadth-first search algorithm (or how to cope with the nondeterminism of reducers),” in Proc. 22nd ACM Symp. on Parallelism in Algorithms and Architectures, 2010, pp. 303–314.

[13] P. Harish and P. Narayanan, “Accelerating large graph algorithms on the GPU using CUDA,” in 14th Intl. Conf. on High Performance Comp., 2007, pp. 197–208.

[14] P. Harish, V. Vineet, and P. Narayanan, “Large graph algorithms for massively multithreaded architectures,” IIIT Hyderabad, Tech. Rep., 2009.

[15] L. Luo, M. Wong, and W. Hwu, “An effective GPU implementation of breadth-first search,” in 47th Design Automation Conference, 2010, pp. 52–55.

[16] S. Hong, T. Oguntebi, and K. Olukotun, “Efficient parallel graph exploration on multi-core CPU and GPU,” in Intl. Conf. on Parallel Architectures and Compilation Techniques, 2011, pp. 78–88.

[17] S. Beamer, K. Asanovic, and D. Patterson, “Direction-optimizing breadth-first search,” in Proc. Supercomputing 2012, 2012, pp. 1–10.

[18] Y. Yasui, K. Fujusawa, and K. Goto, “NUMA-optimized parallel breadth-first search on multicore single-node system,” in Proc. IEEE Intl. Conference on Big Data, 2013, pp. 394–402.

[19] D. Merrill, M. Garland, and A. Grimshaw, “Scalable GPU graph traversal,” in Proc. PPoPP. IEEE, 2012, pp. 117–127.

[20] L.-M. Munguìa, D. A. Bader, and E. Ayguade, “Task-based parallel breadth-first search in heterogeneous environments,” in Proc. HiPC 2012, 2012, pp. 1–10.

[21] A. Buluç and K. Madduri, “Parallel breadth-first search on distributed memory systems,” in Proc. Supercomputing, 2011, pp. 65–79.

[22] H. Lv, G. Tan, M. Chen, and N. Sun, “Understanding parallelism in graph traversal on multi-core clusters,” Computer Science – Research and Development, vol. 28, no. 2-3, 2013, pp. 193–201.

[23] Y. Zhang and E. Hansen, “Parallel breadth-first heuristic search on a shared-memory architectur,” in AAAI Workshop on Heuristic Search, Memory-Based Heuristics and Their Applications, 2006, pp. 1 – 6.

[24] OpenMP API, 4th ed., OpenMP Architecture Review Board, http://www.openmp.org/, Jul. 2013, retrieved: 08.03.2014.

[25] R. Berrendorf, “Trading redundant work against atomic operations on large shared memory parallel systems,” in Proc. Seventh Intl. Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP), 2013, pp. 61–66.

[26] D. Chakrabarti, Y. Zhan, and C. Faloutsos, “R-MAT: A recursive model for graph mining,” in SIAM Conf. Data Mining, 2004, pp. 442 – 446.

[27] C. Groër, B. D. Sullivan, and S. Poole, “A mathematical analysis of the R-MAT random graph generator,” Networks, vol. 58, no. 3, Oct. 2011, pp. 159–170.

[28] DIMACS, DIMACS’10 Graph Collection, http://www.cc.gatech.edu/dimacs10/, retrieved: 08.03.2014.

[29] T. Davis and Y. Hu, Florida Sparse Matrix Collection, http://www.cise.ufl.edu/research/sparse/matrices/, retrieved: 08.03.2014.

[30] J. Leskovec, Stanford Large Network Dataset Collection, http://snap.stanford.edu/data/index.html, retrieved: 08.03.2014.

Multi-Level Queue-Based Scheduling for Virtual Screening Application on Pilot-Agent Platforms on Grid/Cloud to Optimize the Stretch

Bui The Quang, Nguyen Hong Quang
IFI, Equipe MSI; IRD, UMI 209 UMMISCO
Vietnam, Hanoi
e-mails: nguyen.hong.quang@auf.org,
buithequang@gmail.com

Emmanuel Medernach, Vincent Breton
Laboratoire de Physique Corpusculaire
France, Aubière
e-mails: breton@clermont.in2p3.fr,
medernach@clermont.in2p3.fr

Abstract— **Virtual screening has proven a very effective method on grid infrastructures. Operating a dedicated virtual screening platform on grid resources requires optimizing the scheduling policy. The scheduling can be done at 2 levels; at site level and at platform level. Site scheduling is done at each site independently. Each site allocates time slots for different groups of users. Platform scheduling is done at group level: inside a time slot jobs from many users are allocated. Pilot agents are sent to sites and act as a container of actual users jobs. They pick up users jobs from a central queue where the platform scheduling is done. This paper focus on finding platform scheduling policy for pilot-agent platform shared by many virtual screening users. They need a suitable scheduling algorithm at platform level to ensure a certain fairness between users.**

Keywords-Virtual screening; grid computing; scheduling; fairness; stretch; online-algorithm; cloud computing; multilevel queue scheduling; SimGrid.

I. INTRODUCTION

In silico (i.e., computer-assisted) drug discovery [1] offers an efficient alternative to reduce the cost of drug development and to speed-up the discovery process. Virtual screening (VS) is achieved through a pipeline analysis, first step of which requires using a docking software, such as Autodock [2], Dock [3] or FlexX [4] to predict potential interacting complexes of small molecules in protein binding sites. Large scale virtual screening, and especially its docking step, consumes large computing resources. As docking is an embarrassingly parallel process where thousands to millions of compounds are tested *in silico* against a biological target, it was successfully deployed on grid computing to reduce the computation time. Some large scale VS projects in the past have been deployed successfully on grids, such as WISDOM [5][6], and WISDOM-II [8] on malaria, and Avian Flu Data Challenge [7].

Pilot-agent platforms are tools used for submitting and controlling a large number of user jobs on grid infrastructures. Several pilot agent platforms have been developed, such as WPE [8], DIRAC [9], DIANE [10], glideinWMS [11], and PanDA [12]. The DIRAC pilot-agent platform is now available to the users of several multidisciplinary virtual organizations on EGI (the European Grid Initiative) [9]. As many users share the DIRAC pilot-agent platform, it is important to define a scheduling policy to ensure a certain degree of fairness so that users receive a fair share of system resources. The scheduling policies used on the existing pilot-agent platforms on EGI, are respectively FIFO policy in WPE platform and Round Robin policy in DIRAC platform. The VS project has specific properties, such as divisibility in many independent docking tasks, no order of execution constraints and comparable execution time of all docking tasks. In this paper, we focus on evaluating suitable online scheduling policies for the VS application on the pilot-agent platform to improve user's satisfaction in the system.

Our research is also relevant to applications which have the same properties of VS application (divisibility in many independent tasks, no order of execution constraints and comparable execution time of all tasks) on pilot agent platform on grid/cloud. These applications are used in a variety of scenarios, including data mining, massive searches, parameter sweeps [38], simulations, fractal calculations, computational biology [39], and computer imaging [40][41].

This paper is organized as follows. Section 2 describes the problem and our research objectives. Section 3 presents related works. Section 4 introduces our solution based on multi-queue scheduling. Section 5 discusses our results and presents our simulator based on SimGrid [26]. Finally, we conclude in Section 6 and give some perspectives on this research.

II. PROBLEM STATEMENT

In this paper, we evaluate the performances of many scheduling policies applied on the central pool of pilot jobs. The criterion used is the stretch for all users. We, then, explore new scheduling policy based on multiplexing user group queues depending onsome probability parameter. We will first describe our computing platform in details, then we will explain our scheduling policies and evaluate them with the help of simulation based on real workload traces.

A. Pull model, 2-level scheduling and limited machine availability property of scheduling

A pilot-agent platform uses pull model for efficient submission and controlling of user tasks: tasks are no longer pushed through the grid scheduler but are put in a master pool and pulled by pilot agents running on computing nodes. Scheduling job is the process of ordering tasks in this pool. List scheduling is applied in it. The pilot-agent itself is a regular grid job that is started through a grid resource manager. It is automatically submitted by platform and run on a computing machine on grid. We can see a pilot agent as container of jobs. The pull model adapts to heterogeneous property of grid (faster machine will pull more tasks than the other), reduces faults (resubmission of failed tasks) and improves latency (the waiting time of job in grid scheduler is reduced).

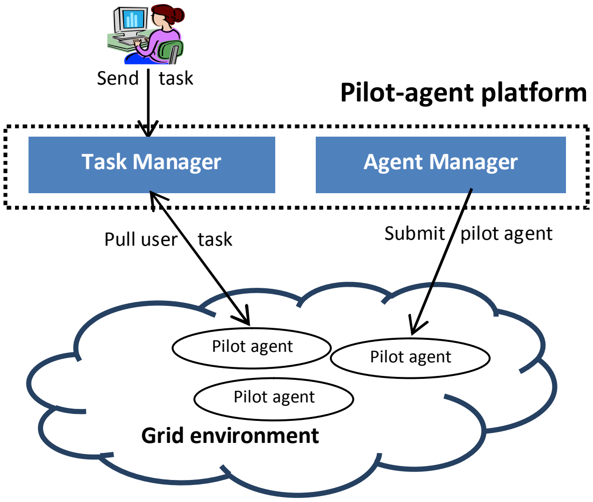


Figure 1. Pull model in pilot agent platform on grid.

As shown in Figure 1, a pilot-agent platform has two main modules, the Task Manager and the Agent Manager.

The pilot agents are submitted automatically to grid by Agent Manager. Then each pilot agent communicates with Task Manager and asks a user task to be executed. Task Manager receives tasks from user and control queue of user tasks. Also, it receives request from pilot agent, choose some task from queue and sends it to pilot agent.

Scheduling of pilot agent platform on grid takes place at site and platform level. The site level scheduling takes place in the site scheduler. Pilot agents sent by the platform are distributed to the sites according to the grid scheduling policy. The platform level scheduling is done by the platform's Task Manager. User sends the VS project to the Task Manager, where docking tasks are put in the task queue. The Task Manager Scheduler calculates task priorities, and responds to pilot agents requests by sending to them tasks ranked with the highest priority. There are underlying grid architectural scheduling and logical scheduling for the specific grid applications. We are concerned with scheduling issue for many virtual screening application users who share the grid resources given to the same group privilege.

Moreover, the platform level scheduling has limited machine availability property. Because each computing center requires some limits to the maximum computing time for grid job, each pilot agent is available for a limited period. Therefore, the number of machines available for the platform changes over time. This specific property makes our analysis directly relevant to cloud infrastructures where users buy computing resources for a limited time.

This paper focuses on finding out the most suitable scheduling policy at platform level to optimize the satisfaction of VS users.

B. The stretch – a measure for user's satisfaction in platform

To work on the fairness of scheduling policy, we need to define a good metric for the satisfaction of an individual user on platform. In the parallel scheduling literature, metrics used to measure the performance of scheduling policies can be classified in two groups: System-centric metrics and Job-centric metrics:

- System-centric metrics to assess platform utilization: C_j denotes the completion time of job j. Makespan (the maximum of the job termination time, $\max_j C_j$) or the sum of completion time ($\Sigma_j C_j$) are common objective functions. Minimization of makespan or sum of completion times is conceptually a system-centric approach.
- Job-centric metrics (Flow time, stretch, etc.) to assess user experience: Flow time F is the time an individual job spends in the platform. The stretch S (also called slowdown) is a particular case of weighted flow time: for job j with size W_j and flow time F_j, the stretch is defined as F_j/W_j. In the context of variable job sizes, the stretch is more relevant to describe user experience than the flow time [24].

This paper focuses on user-centric metric. The stretch is here measured for a group of jobs all belonging to the same user. Assume that user has U jobs, F_j is flow time of job j, W is total size of U jobs, the stretch is then defined as $\max_{j\in U} F_j/W$

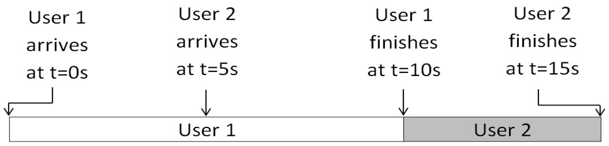


Figure 2. Example of the stretch for two users.

Figure 2 illustrates the mean of stretch in the case of two users sending jobs to a platform. User 1 has a job with size 10 that reaches the platform at t=0s. User 2 has a job with size 5 that reaches the platform at t=5s. User 1 job is executed before user 2 job. As in Figure 2, user 1 finishes at t=10s. Because user 2 has to wait user 1 job completion, he finishes at t=15s. Although the two users spent the same time on the platform (10s), user 2 satisfaction is worse than user 1's. The stretch of user 1 is equal to 10/10=1 while the stretch for user 2 is equal to 10/5=2. This example illustrates why the stretch S measures the user experience on the platform: the larger the stretch, the lower the satisfaction.

Our goal is to identify the best scheduling policy to minimize the stretch of all VS users of a shared pilot-agent platform. We use the max-stretch (S_{max}) metrics to measure of fairness in our scheduling problem. In our latest research [37], we have compared two well-known scheduling policies, Shortest Processing Time (SPT-the user with the least number of tasks has the highest priority) and Longest Processing Time (LPT: the user with the greatest number of tasks has the highest priority), to the scheduling policies currently used on the existing platforms (FIFO and Round Robin). Simulation result and experimentation result on real platform has shown that SPT is the best policy in these 4 policies for online job-centric stretch optimization with virtual screening application on pilot-agent platform on grid/cloud [37].

Although SPT policy is very good online algorithm for optimization of the stretch, this policy has a disadvantage, i.e., it has the tendency to push users with many tasks to the end of the task queue. Sometimes, these users have to wait a long time for a long series of users with less number of jobs. In the worst case, they have to wait forever. Furthermore, research on grid workloads in [27] showed that there are two types of grid users: normal users and data challenge users. Normal user submits to the grid little number of tasks, but the number of user in this group is very large. While data challenge user group submit to the grid very large number of jobs, but the number of users in this group is small. If we use the original SPT policy for all of users in the pilot agent platform, data challenge user will be negatively affected due to large number of user in normal group.

In this paper, we propose a new scheduling policy named SPT-SPT for platform level scheduling in pilot agent platform using multi-level queue scheduling techniques to improve the stretch of VS users. Instead of using one task queue implemented by SPT policy for all users, we use two separate task queues: one for normal user group and another one for data challenge user group. These two task queues are both using SPT policy to optimize the user's stretch in each one. Moreover, a task queue is assigned a parameter p ($p \epsilon [0, 1]$) and the rest one with 1 - p. This parameter is the probability that task queue will be selected by Task Manager when Task Manager receives request from pilot agents. We can see that for $p > 0$, this policy ensures that the data challenge group did not have to wait for the normal group to be entirely empty. Therefore, all users will get an opportunity to utilize grid resources efficiently. The rest of paper is organized as follows.

III. RELATED WORK

A. Grid scheduling

Grid scheduling has been abundantly studied: some surveys of grid scheduling algorithms are proposed in [14][15] and performance of some priority rule scheduling algorithms is presented in [33]. DIET platform [17] is a GridRPC middleware relying on the client/agent/server paradigm. The scheduling on DIET changes from FIFO, Round Robin and CPU-based scheduling. But, the operation of DIET platform is different with pilot-agent platform: DIET use both “push” and “pull” scheduling. Mandatory requests are pushed from clients to resources, whereas optional requests are pulled by resources from clients. Pilot-agent platform takes most scheduling decisions in a centralized agent, in contrast, each client and each server contributes to taking scheduling decisions in DIET. Therefore, the solutions brought by research of scheduling problem on DIET platform are not directly applicable to our problem statement.

Berman et al. [18] presented a scheduling solution in application level called AppLeS. They describe an application specific approach to scheduling individual parallel applications on production heterogeneous systems. They utilize comprehensive information about application and resource to optimize execution time of application on grid. Our goal is not to optimize the execution time of all users but the quality of service for each user.

Existing pilot agent platforms such as DIANE [10], WPE [8], PanDA [12], DIRAC [9] and glideInWMS [11] have different scheduling policies: WPE and DIANE platforms use FIFO while DIRAC uses Round Robin policy. The VS projects have specific properties, such as divisibility in many docking tasks and no order of execution constraints. Therefore, we need to find a suitable online scheduling policy for the VS application on the pilot-agent platform. Fortunately, in some platform, such as DIRAC platform, we can configure the specific scheduling policy for a user group sharing the same application. So, we can apply suitable policy in a VS user group to improve fairness.

B. Cloud scheduling

As mentioned earlier, the limited machine availability property of the scheduling problem on pilot agent platform is similar with scheduling on cloud environment because on cloud environment, user buys some resources with limited duration. When a VS project is deployed on an Infrastructure As A Service (IAAS) cloud, docking task will be executed on a virtual machine with limited availability.

Some researches on cloud scheduling, such as [19][20], have presented their scheduling algorithms on cloud to optimize the speed of resources allocation, the price to pay and the utilization of system resource. But our object is optimization of the fairness of users when they share pilot-agent platform together.

Luckow et al. [21] proposed the design and implementation of a SAGA-based Pilot-Job system, which supports a wide range of application types, and is usable over a broad range of infrastructures from grids/clusters to cloud computing. Fifield et al. [22] showed also an extension of the pilot agent platform DIRAC on cloud computing by submitting pilot agent on Virtual Machine on cloud, such as Amazon EC2. Therefore, our research is also relevant to pilot-agent platforms on Cloud environments.

C. *Scheduling for stretch optimization with limited machine availability constraints*

Many groups have conducted research on optimizing job-centric stretch in the context of dedicated machines (i.e. always available). Muthukrishnan et al. [23] presented the efficiency of the optimal on-line algorithm SPT on uniprocessor and multi-processor. Their objective is optimizing the average of the stretch. Legrand et al. [24] has shown that SPT is quite effective at max-stretch and sum-stretch optimization in problems with continuous machines. But, compared to these studies, our scheduling problem uses a user-centric definition of stretch and adds an additional constraint: machines have limited availability. With this property, the number of machines available for platform changes over time and the complexity of problem increases. Schmidt [16] have reviewed some scheduling algorithm in the context of limited machine availability. LPT is one of the online scheduling algorithms proposed in this research. But these researches are done on system-centric metrics (makespan, sum of completion time, etc.). In our latest research for scheduling for stretch optimization with limited machine availability constraints, we compared two well-known scheduling policies, SPT and LPT, to the scheduling policies currently used on the existing platforms (FIFO and Round Robin). Simulation result and experimentation result on real platform showed that SPT policy is the best policy in these 4 policies for optimization of user stretch in the context of limited machine availability.

D. *Multi-level queue scheduling*

Various algorithms for multilevel queue are discussed in [34] to improve different CPU scheduling factors as turnaround time, waiting time, starvation problem, etc. These researches are done on multi-level queue scheduling technique used for CPU scheduling in operating system in a computer. In contrast, grid is a distributed computing environment. User tasks are executed by many distributed pilot agent on grid. Therefore, we need to evaluate multi-level queue technique on distributed computing environment of grid computing.

Chouhan et al. [35] and Kumaresh et al. [36] presented a scheduling policy for grid computing using multilevel feedback queue scheduling technique and multilevel queue scheduling to avoid the starvation of low priority jobs in the global scale of grid. However, in our context we need to find out a policy for platform level scheduling of pilot agent platform. And our objective is minimization of the user stretch, a measure for user experience.

In conclusion, to the best of our knowledge, no research on optimizing user stretch was conducted in the case of limited machine availability. In the next section, we describe solution proposed and our simulator used to evaluate and compare the performance of new policy to original SPT scheduling policy.

IV. Solution proposed

In this section, we briefly explain the proposed solution using multilevel queue technique in Task Manager of pilot agent platform. Administrator of pilot agent platform creates two groups for VS users: Normal group and Data Challenge group. VS user is assigned to Normal group by default. When someone needs to process a big virtual screening project, he will contact with administrator of pilot agent platform to change his role to Data Challenge group in some days or some weeks.

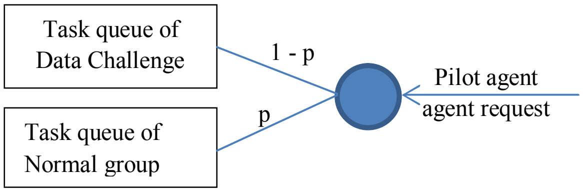


Figure 3. SPT-SPT policy with two task queues.

In the Task Manager module of the platform, we build two separate task queues: one queue for normal group and another one for data challenge group. Task queue of normal group is assigned priority p and task queue of data challenge group is assigned priority 1 – p. These indices are the probability that task queue is chosen by Task Manager to send pilot agent their task when Task Manager receives request from pilot agent.

According to our latest research, SPT is better than FIFO, LPT, RR for minimizing the user stretch. Therefore these both task queues use SPT policy for optimizing the stretch of user on each one (We tried with policy SPT-RR: SPT on normal group task queue and Round Robin in data challenge task queue and SPT-FIFO: SPT on normal group task queue and FIFO on data challenge task queue. The result of SPT-RR and SPT-FIFO is worse than SPT policy). We use algorithm SPT-SPT to control user task in Task Manager as described in Algorithm 1. The platform administrator can change value of parameter p in the configuration of pilot agent platform.

Algorithm 1: SPT-SPT policy in Task Manager scheduler

```
INPUT:
  p = normal user group task queue parameter
  Pilot agent requests are received online
OUTPUT:
  Scheduling of tasks to pilot agents

for all pilot agent request received
do
  if empty(data challenge task queue)
     AND empty(normal task queue)
  then
    Push pilot agent to pilot agent queue
  else if empty(data challenge task queue)
  then
    Send task of normal task queue to pilot agent
  else if empty(normal task queue)
  then
```

```
    Send task of data challenge task queue to pilot agent
  else
    if ( random(0,1) < p )
    then
      Send task of normal task queue to pilot agent
    else
      Send task of data challenge task queue to pilot agent
    end if
end if
```

This is an online scheduling algorithm and it is linear in time complexity. We can see that with $0 < p < 1$ there are always pilot agents taking a task from Data Challenge group. Therefore, Data Challenge user does not have to wait for a long series of normal users having less tasks. We will use our simulator to find out the best value of p to decrease $S_{\max}$ of Data Challenge group and do not increase very much $S_{\max}$ of Normal group.

V. EXPERIMENTATION ON SIMULATOR

A. Scenario description

There are three parameters for our experimentation: Configuration of grid infrastructure, VS user workload and parameter for pilot-agent platform.

1) Configuration of grid infrastructure

To simulate realistically the operation of a pilot-platform, we used archives of the AuverGrid regional multidisciplinary grid infrastructure in Auvergne (France) in 2004-2005, available on the Grid Workload Archive site. The AuverGrid infrastructure in this period is detailed in Table 1, including machines relative speeds. All machines in a cluster have the same speed.

TABLE I. CONFIGURATION OF THE AUVERGRID INFRASTRUCTURE

Cluster name	Number of Worker Node	Relative speed of Worker Node	Limitation of computing time (second)
CLRLCGCE01	112	1	258220
CLRLCGCE02	84	1.1	258220
CLRLCGCE03	186	1.6	258220
IUT15	38	0.8	172800
OPGC	55	1.4	172800

2) Virtual screening user workload

According to the research on the real grid workload done by Medernach [27], we use their model to generate workload example for virtual screening user as below:

Normal group has X^{normal} users, for each user U_i^{normal} in this group (ref. section 3.1: Mathematical model):

- $N_{j(i)}^{normal}$, the number of docking tasks of project j submitted by user i, is generated by a Geometric random distribution : $\gamma = a \times b^i$, parameter a corresponds to the first user mean number of docking tasks and parameter b is the geometric progression.
- $[r_{j(i)}^{normal}, r_{j+1(i)}^{normal}]$, the interval between submissions of two projects consecutive of user i, is generated within a Poisson random distribution with parameter $\lambda = c \times d^i$, parameter c corresponds to the first user mean inter-arrival time and parameter d is the geometric progression.
- We require max_time = 400 seconds to generate VS user workload example: $r_{j(i)}^{normal}$ < max_time

The same model is used for generating workload for data challenge group. In our simulation, we used the following parameters:

- Normal user group has parameters:

  $a = 1, b = d = \sqrt[40]{20}, c = 600$
- Data Challenge group has parameters:

  $a = 60000, b = d = \sqrt[20]{10}, c = 30000$

The workloads of normal user and data challenge user are combined in VS workload example. We generated 500 VS workload examples for each dataset. There are 4 datasets: case 00, case 01, case 02 and case 03 with different numbers of users in each group as table 2.

TABLE II. NUMBER OF USER IN EACH GROUP ON DATASET

	Number of Normal users	Number of Data Challenge users	Max time (second)
Case 00	119	1	400.000
Case 01	195	5	400.000
Case 02	190	10	400.000
Case 03	185	15	400.000

B. Simulation result and analysis

For each dataset (from case 00 to case 03), we run simulation on 500 VS workload examples for FIFO policy, SPT policy and SPT-SPT policy with the percentage of pilot agent for normal group p = 0%, 10%, 30%, 50%, 70%, 90% and 100% (ref. Algorithm 1). We calculate the $S_{\max}^{DC}$ and $S_{\max}^{normal}$ on each VS workload example as formula 1 and 2. Next, we figure out the average of $S_{\max}^{DC}$ and $S_{\max}^{normal}$ in each dataset. Figure 4 presents simulation results for case 00, case 01, case 02 and case 03. The more percentage of pilot agents for normal group, the more grid resource is reserved for normal user group and the less grid resource for data challenge group. Therefore, we can see in Figure 4 that, when p increases, the $S_{\max}^{normal}$ decreases and the $S_{\max}^{DC}$ augments. From the result of case 00, case 01, case 02 and case 03, we chose p = 70%, where the max-stretch of normal user group changes a little but the max-stretch of data challenge user decreases very much in comparison with original SPT policy. With this value of p, SPT-SPT policy improves user experience compare to original SPT policy. The best value of p depends on the number of task of two groups. In SPT-SPT policy, administrator of pilot agent platform can adjust this parameter according to the actual situation for optimizing the stretch of two groups.

Table 3 presents the number of users of group in each dataset, the value $S_{\max}^{normal}$ and $S_{\max}^{DC}$ in SPT-SPT policy with p = 70%, in the original SPT policy and in FIFO policy. We can see that, in all cases the $S_{\max}^{normal}$ and $S_{\max}^{DC}$ in FIFO policy are higher than this one in SPT and SPT-SPT policy. For example, a normal user arrives just after a data challenge user, he has to wait very long time, so that S_{max} is very large in FIFO policy. The result shows that FIFO policy is not good to minimize the user stretch. Comparison between SPT and SPT-SPT policy (with p=70%), we can see that $S_{\max}^{normal}$ is approximate but $S_{\max}^{DC}$ in SPT-SPT policy is smaller than SPT policy. Moreover, from case 00 to case 03, we can see that the more data challenge users, the more $S_{\max}^{DC}$ is smaller than this one in SPT policy. This means that in the context with many data challenge users, the SPT-SPT policy is very much better than SPT policy.

VI. CONCLUSION AND PERSPECTIVE

The paper described a new scheduling policy for virtual screening application on pilot agent platform for optimizing the stretch of user. We proposed SPT-SPT policy using multilevel queue technique for platform level scheduling on pilot agent platform. This approach, based on the research of grid workload, has shown that there are two types of users : many users submitting small number of tasks and a little number of users submitting a large number of tasks. Simulation results showed that SPT-SPT policy (with 70% of pilot agent reserved for normal user group and 30% of pilot agent reserved for data challenge group) has better result on user stretch than original SPT policy. The stretch of data challenge user group decreases and the stretch of normal user is almost unchanged.

Infrastructure as a Service cloud is similar to our problem with limited availability of pilot agent on grid because their users buy access to computing resources for a limited time. Therefore, we also propose to implement SPT-SPT in deployment of virtual screening application on cloud environments.

ACKNOWLEDGMENT

We are grateful to “Agence Universitaire de la Francophonie” for their grant support. The authors are also thankful to France Grilles and EGI for providing computing resources used on the Biomed Virtual Organization. We also acknowledge FKPPL and FVPPL LIAs support for travel and exchange between research groups in France, Korea and Vietnam. We warmly thank Vanessa Hamar and Andrei Tsaregorodtsev for their assistance with the DIRAC platform.

REFERENCES

[1] V. S. Rao and K. Srinivas,. “Modern drug discovery process: an in silico approach.”, Journal of Bioinformatics and Sequence Analysis, 2(5), 2011, pp. 89-94.

[2] D. S. Goodsell, G. M. Morris, and A. J. Olson, “Automated docking of flexible ligands: applications of AutoDock.”, Journal of Molecular Recognition, 9(1), 1996, pp. 1-5.

[3] R. G. Coleman and K. A. Sharp, “Protein pockets: inventory, shape, and comparison,” Journal of chemical information and modeling, 50(4), 2010, pp. 589-603.

[4] I. Schellhammer and M. Schellhammer, “FlexX-Scan: Fast, structure-based virtual screening,” PROTEINS: Structure, Function, and Bioinformatics, 57(3), 2004, pp. 504-517.

[5] N. Jacq, V. Breton, H. Y. Chen, L. Y. Ho, M. Hofmann, H. C. Lee, and M. Zimmermann, “Large scale in silico screening on grid infrastructures,” [arXiv preprint cs/0611084].

[6] N. Jacq, J. Salzemann, F. Jacq, Y. Legré, E. Medernach, J. Montagnat, and V. Breton, “Grid-enabled virtual screening against malaria,” Journal of Grid Computing, 6(1), 2008, pp. 29-43.

[7] H. C. Lee, J. Salzemann, N. Jacq, H. Y. Chen, L. Y. Ho, I. Merelli, and Y. T. Wu, “Grid-enabled high-throughput in silico screening against influenza A neuraminidase,”, IEEE transactions on nanobioscience, 2006, pp. 288-295.

[8] V. Kasam, J. Salzemann, M. Botha, A. Dacosta, G. Degliesposti, R. Isea, D. Kim, A.Maass, C. Kenyon, G. Rastelli, M. Hofmann-Apitus and V. Breton, “WISDOM-II: Screening against multiple targets implicated in malaria using computational grid infrastructures,” Malaria Journal, 2009, 8(1), pp. 88-103,

[9] E. van Herwijnen, J. Closier, M. Frank, C. Gaspar, F. Loverre, S. Ponce, and M. Gandelman, “Dirac—distributed infrastructure with remote agent control,” Conference for Computing in High-Energy and Nuclear Physics (CHEP 03), 2003.

[10] J. T. Mościcki, “Distributed analysis environment for HEP and interdisciplinary applications,” Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 502(2), 2003, pp. 426-429.

[11] I. Sfiligoi, “glideinWMS—a generic pilot-based workload management system,” Journal of Physics: Conference Series, vol. 119, No. 6, IOP Publishing, Jul. 2008

[12] T. Maeno, “PanDA: distributed production and distributed analysis system for ATLAS,” Journal of Physics: Conference Series, vol. 119, No. 6, IOP Publishing, 2008

[13] R. F. Da Silva, S. Camarasu-Pop, B. Grenier, V. Hamar, D. Manset, J. Montagnat, and T. Glatard, “Multi-infrastructure workflow execution for medical simulation in the Virtual Imaging Platform,” Proceedings of the 9th HealthGrid Conference. 2011, pp. 1-10.

[14] D. Maruthanayagam and R. Uma Rani, “Grid scheduling algorithms: a survey,” International Journal of Current Research. vol. 11 , Dec. 2010, pp. 228-235.

[15] C. Jiang, C. Wang, X. Liu, and Y. Zhao, “A survey of job scheduling in grids,” Advances in Data and Web Management, Springer Berlin Heidelberg, 2007, pp. 419-427.

[16] G. Schmidt, “Scheduling with limited machine availability”, European Journal of Operational Research, 121(1), 2000, pp. 1-15.

[17] P. Marrow, E. Bonsma, F. Wang, and C. Hoile, “DIET—a scalable, robust and adaptable multi-agent platform for information management,” BT technology journal, 21(4), 2003, pp. 130-137.

[18] F. Berman, R. Wolski, S. Figueira, J. Schopf, and G. Shao, “Application-level scheduling on distributed heterogeneous networks,” Proceedings of Supercomputing. vol. 96. 1996, pp. 1-28.

[19] S. Pandey, L. Wu, S. M. Guru, and R. Buyya, “A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments,” AINA '10: Proceedings of the 2010, 24th IEEE International Conference on Advanced Information Networking and Applications. Washington, DC, USA, IEEE Computer Society, 2010, pp. 400-407.

[20] W. Li, J. Tordsson, and E. Elmroth, "Modeling for dynamic cloud scheduling via migration of virtual machines," Proceedings of the 3rd IEEE International Conference on Cloud Computing Technology and Science (CloudCom 2011), 2011, pp. 163-171.

[21] A. Luckow, L. Lacinski, and S. Jha, "SAGA BigJob: An extensible and interoperable pilot-job abstraction for distributed applications and systems," Cluster, Cloud and Grid Computing (CCGrid), 10th IEEE/ACM International Conference, 2010, pp. 135-144.

[22] T. Fifield, A. Carmona, A. Casajús, R. Graciani, and M. Sevior, "Integration of cloud, grid and local cluster resources with DIRAC", Journal of Physics: Conference Series, vol. 331, No. 6, 2011. [ref. 062009, doi:10.1088/1742-6596/331/6/062009]

[23] S. Muthukrishnan, R. Rajaraman, A. Shaheen, and J. E. Gehrke, "Online scheduling to minimize average stretch," IEEE Symposium on Foundations of Computer Science, 1999, pp. 433-442.

[24] A. Legrand, A. Su, and F. Vivien, "Minimizing the stretch when scheduling flows of biological requests," Proceedings of the eighteenth annual ACM symposium on Parallelism in algorithms and architectures, 2006, pp. 103-112.

[25] B. Chen, C. N. Potts, and G. J. Woeginger, "A review of machine scheduling: Complexity, algorithms and approximability" Handbook of combinatorial optimization, ch 3, 1998, pp. 21-169.

[26] H. Casanova, A. Legrand, and M. Quinson, "SimGrid: a generic framework for large-scale distributed experiments" Proceeding 10th International Conference Computer Modeling and Simulation, Mar. 2008, pp. 126-131

[27] E. Medernach, "Workload analysis of a cluster in a grid environment" Job scheduling strategies for parallel processing, Springer Berlin Heidelberg. 2005, pp. 36-61.

[28] E. L. Lawler, J. K. Lenstra, and A. R. Kan, "Sequencing and scheduling: Algorithms and complexity," Handbooks in operations research and management science, 4, 1993, pp. 445-522.

[29] T. C. E. Cheng and C. C. S. Sin, "A state-of-the-art review of parallel-machine scheduling research," European Journal of Operational Research, 47(3), 1990, pp. 271-292.

[30] Jain, Raj, "The art of computer systems performance analysis," vol. 182. Chichester: John Wiley & Sons, 1991.

[31] A. B. Downey, "A parallel workload model and its implications for processor allocation," Cluster Computing 1.1, 1998, pp. 133-145.

[32] Feitelson, G. Dror "Packing schemes for gang scheduling," Job Scheduling Strategies for Parallel Processing, Springer Berlin Heidelberg, 1996.

[33] Z. R. M. Azmi, K. A. Bakar, A. H. Abdullah, M. S. Shamsir, and W. N. W. Manan, "Performance Comparison of Priority Rule Scheduling Algorithms Using Different Inter Arrival Time Jobs in Grid Environment," International Journal of Grid and Distributed Computing, 4(3), 2011, pp. 61-70.

[34] C. Vaishali. and R. Supriya "A Review of Multilevel Queue and Multilevel Feedback Queue Scheduling Techniques," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, iss. 1, pp. 110-113.

[35] D. Chouhan, S. M. Dilip Kumar, and B. P. Vijaya Kumar, "Multilevel Feedback Queue Scheduling Technique for Grid Computing Environments", in Proceedings of International Conference on Advances in Computing SE - 1, vol. 174, A. Kumar M., S. R., and T. V. S. Kumar, Eds. Springer India, 2012, pp. 1–7.

[36] V.S Kumaresh,S. Prasidh, B. Arjunan,S Subbhaash and M.K. Sandhya, "Multilevel Queue-Based Scheduling for Heterogeneous Grid Environment", International Journal of Computer Science Issues, vol. 9, Issue 6, No 3, Nov. 2012, Springer India, 2012.

[37] T. Q. Bui, E. Medernach, V. Breton, H. Q. Nguyen, and Q. L. Pham, "Stretch Optimization For Virtual Screening on Multi-user Pilot-agent Platforms on Grid/Cloud", Proceedings of the Fourth Symposium on Information and Communication Technology, 2013.

[38] D. Abramson, J. Giddy and L. Kotler. "High Performance Parametric Modeling with Nimrod/G: Killer Application for the Global Grid," IPDPS'2000, Cancun Mexico, IEEE CS Press, 2000, pp. 520-528.

[39] J. R. Stiles, T. M. Bartol, E. E. Salpeter, and M. M. Salpeter, "Monte Carlo Simulation of Neuromuscular Transmitter Release Using MCell, a General Simulator of Cellular Physiological Processes," Computational Neuroscience, 1998, pp. 279-284.

[40] S. Smallen, W. Cirne and J. Frey et al, "Combining Workstations and Supercomputers to Support Grid Applications: The Parallel Tomography Experience," Proceeding of the HCW'2000-Heterogeneous Computing Workshop, 2000

[41] S. Smallen, H. Casanova, and F. Berman, "Applying Scheduling and Tuning to On-line Parallel Tomography," Proceedings of Supercomputing 01, Denver, Colorado, USA, Nov. 2001

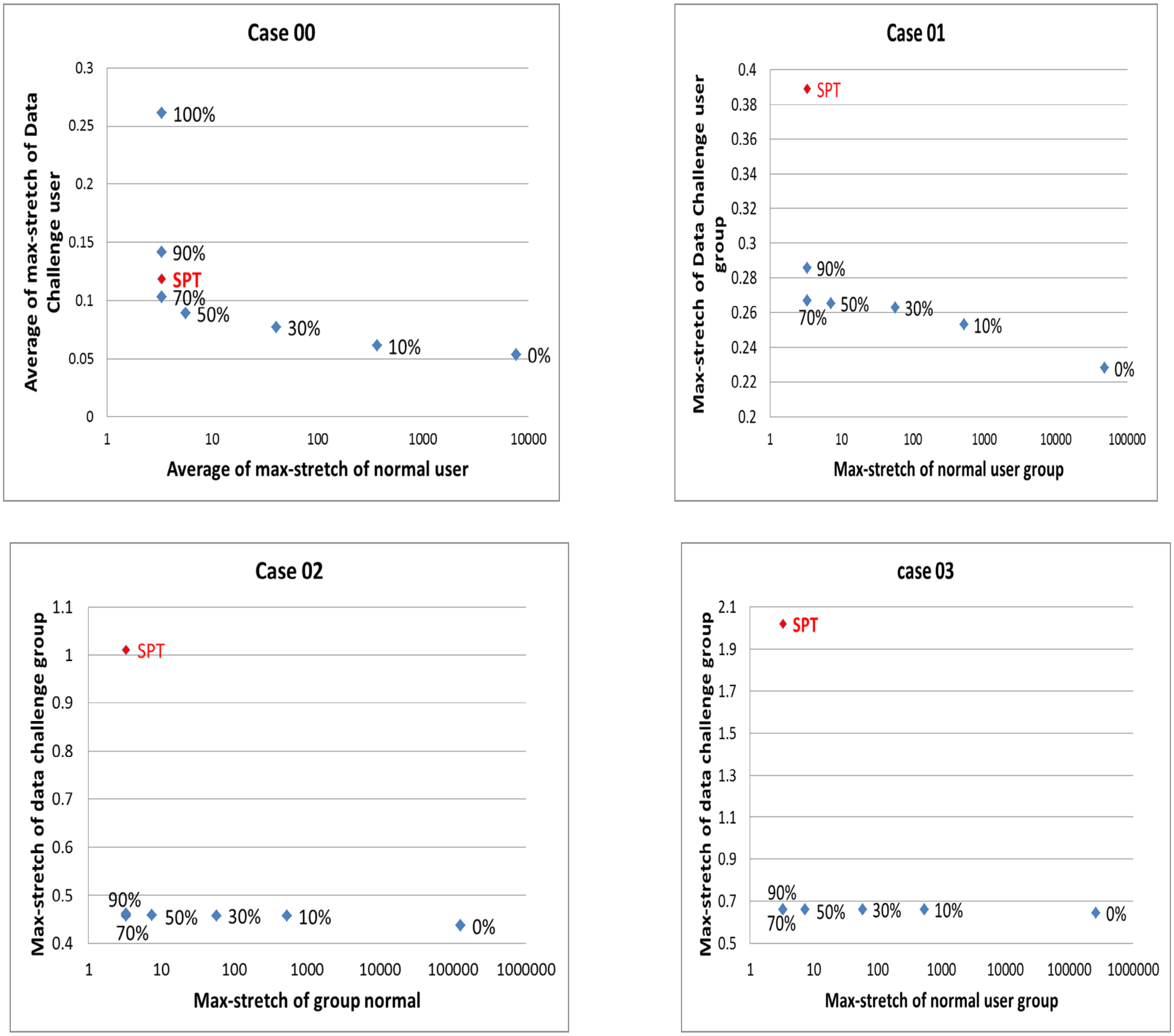


Figure 4. Average of max-stretch of two groups versus % pilot agent for normal user group in case 00, 01, 02, 03

Power-aware Work Stealing in Homogeneous Multicore Systems

Shwetha Shankar
Intel Corporation
Austin, Texas, USA
shwetha.shankar@intel.com

Greg LaKomski, Claudia Alvarado, Richard Hay,
Christopher Hyatt, Dan Tamir, and Apan Qasem
Department of Computer Science,
Texas State University,
San Marcos, Texas, USA
{gl082,ca1015,rh1375,ch1662,dt19,apan}@txstate.edu

Abstract— **Excessive power consumption affects the reliability of cores, requires expensive cooling mechanisms, reduces battery lifetime, and causes extensive damage to the device. Hence, managing the power consumption and performance of cores is an important aspect of chip design. This research aims to achieve efficient multicore power monitoring and control via operating system based power-aware task scheduling. The main objectives of power-aware scheduling are: 1) lowering core's power consumption level, 2) maintaining the system within an allowable power envelope, and 3) balancing the power consumption across cores; without significant impact on time performance. In previous research we have explored power-aware task scheduling at the single core level referred to as intra-core scheduling. This paper reports on a research on a power-aware form of inter-core scheduling policy referred to as work stealing. Work stealing is a special case of task migration, where a "starving" core attempts to steal tasks from a "victim", i.e., a "loaded" core. We have performed experiments with ten variants of the work stealing that consider both the power and the performance attributes of the system in the process of selecting a victim core. The experiments conducted show that the power-aware inter-core stealing policies have high potential for power efficient task scheduling with tolerable effect on performance.**

Keywords-task scheduling; task migration; work stealing; power-aware task scheduling; energy delay product

I. INTRODUCTION

Power consumption is a dominant obstacle for performance improvements in the very large scale integration (VLSI) technology. Excessive power consumption affects the reliability of cores. High power dissipation results in high heat generation. This in turn, requires costly cooling mechanisms, affects battery lifetime, and causes damage to semi-conductor devices. Hence monitoring the power consumption is of high importance in the semiconductor industry [1]-[5]. This study aims to address power management issues by concentrating on scheduling techniques available at the Operating System (OS) level [6]-[10].

Task scheduling in a multicore system is composed of three components: task matching, intra-core task scheduling, and inter-core task migration. Task matching is the assignment of new tasks to cores. Intra-core task scheduling policies concentrate on selecting a ready task for execution at the single core level while inter-core task migration policies focus on moving ready tasks between cores. Work stealing, a specific type of task migration, is a multicore scheduling algorithm that can improve performance and achieve efficient dynamic load-balancing [3][5]. In the classical work-stealing environment, cores that are executing tasks are referred to as workers while idle cores are potential thieves (or stealers). Depending on the state (working or idle) cores make choices with regard to available tasks. Each worker must choose the next task to be executed. If an idle core becomes a thief, it must choose the victim core and the task to steal [3]-[5][12]-[19].

Maintaining a homogeneous multicore system within an allowable power envelope and/or balancing the power consumption across cores without drastically affecting performance are the main problems addressed in this paper. The main objective is to devise an efficient power-aware multicore OS task scheduler so that both execution and power consumption of the task are taken into consideration. In addition, this study aims to find mechanisms to lower a core's power consumption and support hot-spot elimination. These objectives are achieved by integrating power characteristics into inter-core work stealing policies.

There is significant amount of research on scheduling algorithms involving execution time as the optimization criteria, focusing on real-time applications, and interacting with hardware [1][2][9]-[11]. However, research on power-aware task scheduling strategies that focus on power consumption issues and integrate power and performance metrics in the scheduling optimization criteria has considerable opportunities for extension. This study incorporates both execution time and power considerations into the OS based task scheduling on homogeneous multicore systems.

The main contribution of this study is the introduction of power efficient inter-core work stealing policies that significantly reduce the energy consumption variance across cores and produce a noticeable improvement in the completion time for different workload scenarios.

This paper is organized in the following way. Section II provides a brief overview of relevant background information. Section III provides details concerning work stealing mechanisms and Section IV includes a review of literature related to research conducted. The literature survey shows that significant research is yet to be done and provokes studies seeking cost-effective power efficient OS task scheduling policies for single and multicore systems. Section V provides details on the experimental setup used to evaluate the devised power efficient policies. Section VI presents details of the set of experiments conducted. Section VII includes evaluation of the experimental results. Finally, Section VIII provides conclusions and proposals for future research.

II. TASK SCHEDULING

In general, task scheduling in multicore systems is done by the OS. On the other hand power management, mainly through scaling of the frequency of cores via Dynamic Voltage and Frequency Scaling (DVFS), is done by firmware. Moreover, there is a significant amount of information concerning execution parameters and power performance that is readily available at the firmware-level, but is not readily available to the OS. Hence, there is semantic gap between the OS and the firmware. Figure 1 shows an ideal situation where the OS and the firmware have extensive hand-shaking utilized for power-aware scheduling and management by the OS. The hand shaking enables the OS to change the working frequency and states of cores. Furthermore, firmware-level execution information is used by the OS for improved scheduling decisions.

Many of the terms related to task matching and scheduling are overloaded. To remove ambiguity, we define the following terms [5]: Task Matching is the process of assigning incoming tasks to processing cores in order to optimize a given metric such as throughput (other terms for this operation are: task scheduling, task mapping, and task distribution). Intra-core task scheduling refers to the scheduling of tasks assigned to a core on that core. Task migration means moving tasks from the ready queue of one core to the ready queue of another core (e.g., work stealing). Often, task migration is referred to as task redistribution.

The Energy Delay Product (EDP) is a performance measure that takes execution time and power into account. The EDP is defined to be $EDP = T \times E = T^2 \times P$, where T denotes the execution time of a task, E is the energy, and P is the average power consumed by the task throughout the execution [1]-[5].

The Highest Response Ratio Next (HRRN) is an intra-core scheduling measure that ranks tasks based on the equation $HRRN = \frac{w+s}{s}$. In this formula, s denotes the remaining task service time, and w is defined to be the amount of time the specific task has been waiting in any system queue.

The Highest Energy-delay-product based Cost Function (HECN) is a power-aware heuristic developed by our research team. It integrates the HRRN scheduling policy and EDP into the scheduling selection criteria. Several versions of the heuristics are used in our experiments. In this research we use the following version: $HECN = \frac{w+EDP}{EDP} = \frac{w+s^2P}{s^2P}$. Hence, the remaining EDP replaces the remaining execution time. Although, in our general matching, scheduling, and task migration framework, HECN is also used for intra-core task scheduling and task matching, this paper concentrates on task migration via work stealing. Task matching and task scheduling are not elaborated upon further. The reader is referred to [5] for further details on these topics. The HECN heuristics involves elements of different physical dimensions. Nevertheless, being a heuristic, this does not pose an issue.

Figure 2 shows the research and simulation framework through a snapshot of an exploratory simulator developed in this research. Figure 3 shows potential patterns of task generation (arrival). These topics are detailed in Section IV. The main components of the framework are described here.

A. Task Matching

Optimal task matching is an assignment of newly generated tasks to available cores that optimizes a cost function such as performance, power consumption, and thermal envelope [1][2][4]. Task matching is an NP complete problem [4]. Hence, numerous heuristics have been developed for finding a sub-optimal solution for the problem. Generally, these heuristics are referred to as matching polices. This topic is out of the scope of the current paper.

B. Intra-core Task Scheduling

Intra-core task scheduling is a scheduling component of single core and multicore systems. The process involves selecting the next task to be executed on a core from the current tasks allocate to that core. Numerous methods addressing different scenarios in preemptive and non preemtive operating systems, including round robin, first come first serve, and HRRN have been explored and implemented [1]-[3][5][9]-[13]. In previous research we have identified the HRRN and its power-aware heuristic variant HECN as the most promising approach for intra-core scheduling [3][5]. Again, this topic is out of the scope of the current paper which concentrates on task migration. Nevertheless, in our simulations, it is assumed that HECN is used for intra-core scheduling. Additionally, HRRN and HECN are used in work stealing decisions.

C. Inter-core Task Migration

Task migration occurs if the system is in extreme imbalance and certain cores experience an extremely high peak in a given parameter while other cores experience an extremely low peak in that parameter.

Under task migration policies, tasks are reallocated to cores. Classification of task migration policies includes 5 main parameters: 1) the trigger for reallocation, 2) the reallocation source core (or cores), 3) the destination core(s), 4) task selection criteria (which affects the set of tasks that are candidates for reallocation) and the set of tasks that are eventually migrated, and 5) the amount of available knowledge concerning the system's state. Work stealing, detailed in Section III, is a special form of task migration. In this case, the trigger for reallocation is the starvation of one or more cores, the source cores are cores that are considered to be loaded (see Section III) and the destination cores are the starving cores. Several core/task selection policies can be considered.

D. Core State Control

The firmware can control the state of a core and place it in several sleep-modes. Additionally, the firmware can control the frequency of cores. The current trend is to equip the OS with this type of control capabilities and this is reflected in our simulation framework depicted in Figure 2.

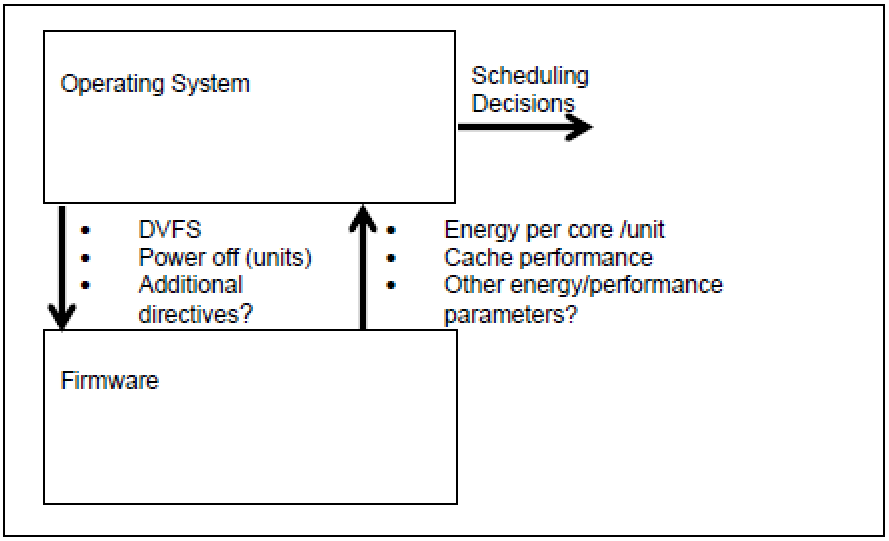


Figure 1. Desired OS and hardware interaction.

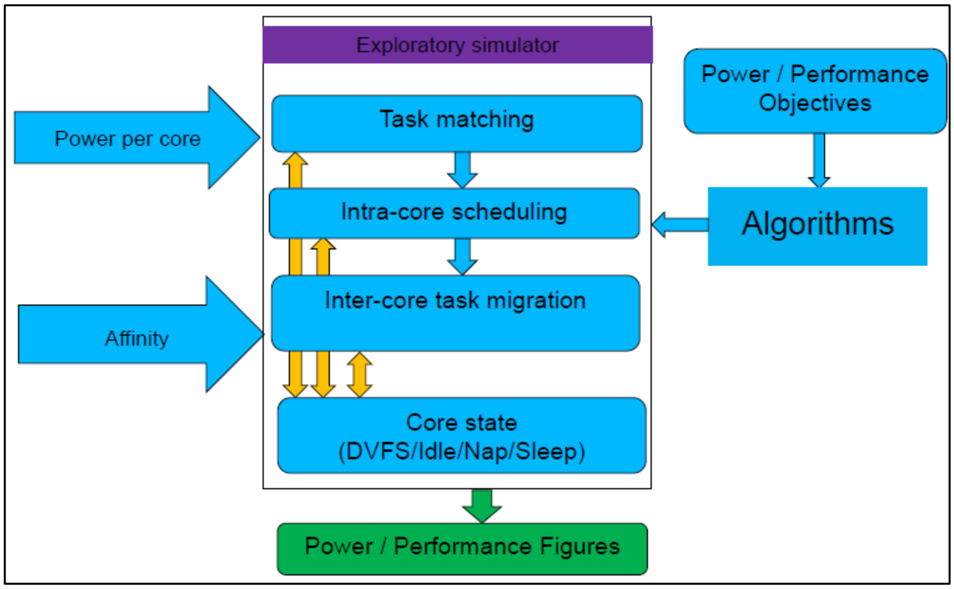

Exploratory simulator
Power per core
Task matching
Intra-core scheduling
Inter-core task migration
Core state (DVFS/Idle/Nap/Sleep)
Affinity
Power / Performance Objectives
Algorithms
Power / Performance Figures

Figure 2. A snapshot of the research and simulation framework developed in this project.

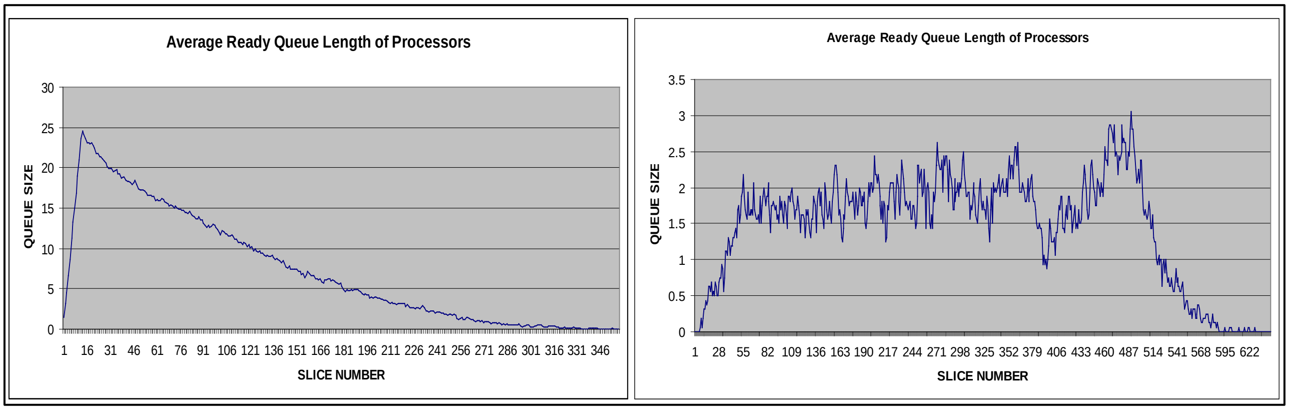


Figure 3. Task arrival modes (a) early saturation mode (b) Steady state task arrival modes

III. POWER-AWARE WORK STEALING

In a multicore system, task migration refers to moving tasks from the ready queue of one core to the ready queue of another core in order to improve performance metrics. This paper concentrates on one type of migration referred to as work stealing [3].

Cores that experience extreme (high or low) imbalance values of a given parameter might initiate a task migration transaction. In this study, the ready queue size is considered as the parameter that indicates imbalance. A core is considered as starved if the number of tasks in its ready queue falls below a threshold T_s. On the other hand, a core is considered as loaded if the number of tasks in the ready queue is above a threshold T_l . A core is considered as normal if it is neither starving nor loaded. This type of core does not participate in work stealing.

A starving core is a potential stealer and a loaded core is a potential victim of stealing. A stealer initiates the stealing process by seeking a victim. The stealer identifies a victim. The victim volunteers a task to be stolen. The stealer steals this task by migrating it to its own ready queue. This process is referred to as work stealing. In a homogeneous multicore system, there is no architecture difference between cores. Hence, all the stealers and all the potential victims can only be distinguished based on execution parameters and not on architecture parameters. This research report concentrates on homogeneous multicore systems.

The process of work stealing involves three steps. The first step is identifying starving and loaded cores. Next, a specific victim has to be selected from the loaded cores. Finally, a specific task to be migrated from the victim to the stealer has to be identified. There are numerous variants and options related to each of these steps. One consideration is the amount of knowledge available to cores. Under the local knowledge model, each core is only aware of its own current status [3][5]. In the global knowledge model, there is an entity (e.g., the OS) that has and utilizes knowledge about the status of each core [15]-[20]. The selection of the victim core and the migrated task can be done in a way that optimizes performance objectives. For example, if there is more than one potential victim, the OS might choose the most loaded core as the victim and the task with the highest wait time in the ready queue of that core as the task for migration.

Traditionally, work stealing has been applied under performance optimization criterion. For example, the stealing decisions (choosing the victim core and the task to be migrated) might attempt to optimize wall to wall time of an entire workload. In this research, the stealing decisions incorporate power and performance objectives. Three main objectives were considered 1) lowering a core's power consumption level, 2) maintaining the system within an allowable power envelope, and 3) balancing the power consumption across cores; without significant impact on time performance. In each set of experiments, one or more of these goals was used in the process of victim and migrated task selection. For example, in order to achieve balance in power consumption, the loaded core that has consumed the most amount of energy in the last K time slices is most likely to be selected as one of the victims. Additional considerations include the amount of global knowledge assumed, affinity between tasks and cores (e.g., due to recent use of the core cache), and the power consumption characteristics of the tasks in the ready queue of potential victims.

IV. LITERATURE REVIEW

This section discusses the relevant research available on single and multicore task scheduling policies that consider the energy consumption of cores.

Kashif et al. and Kim et al. propose a Priority-based Multi-level Feedback Queue Scheduler (PMLFQS) for mobile devices [11][12]. Their papers, however, focus on the firmware role rather than the OS role in power management.

Wu et al. propose Low Thermal Early Deadline First (LTEDF), a temperature-aware task scheduling algorithm for real-time multicore systems [13]. If cores are thermally saturated, task migration is performed to alleviate the saturation. The paper is focused on real-time systems and on lowering the peak power and temperature consumptions. Our study, however, concentrates on general applications. Moreover, rather than limiting the consideration to peak power, this research considers balancing the power consumption across cores in the system.

Zhou et al. propose an algorithm referred to as THRESHHOT [14]. At each step, THRESHHOT selects the hottest task that does not exceed the thermal threshold using an online temperature estimator, leveraging the performance counter-based power estimation. The paper, however, focuses on batch processes on a single core and is intended to lower final core temperature. Our study aims to consider varying type of processes (beyond batch processes) on a multicore system.

Quintin et al. detail the Classic Work Stealing Algorithm. [15]. In addition, they propose the idea of grouping cores as *Leaders* or *Slaves* and restricting the stealing according to the grouping. In our research, stealing policies are devised for a homogeneous system such that all cores (that have load imbalance) can participate in stealing with the help of one efficient central unit.

Guo et al. propose two policies that fit high granular parallel processing environment [16]. Our work aims at developing power-aware policies for all types of workload including high and low granularity parallelism workloads.

Agarwal et al. propose a Central Task Scheduler that can maintain information of all the cores in the system [17]. Sudarshan et al. discuss a similar policy that mainly consists of a dispatcher and nodes [18]. Our research considers

ISBN: 978-1-61208-339-1

several levels of information sharing from local to global knowledge sharing.

Robison et al. propose considering task to core affinity as a part of the task matching. They use a “Mailbox” and FIFO mechanism to handle the affinity [19]. Our research involves affinity at all levels of scheduling, not only at the matching stage.

Faxén et al. suggest Sampling Victim selection where a thief samples several potential victims and selects the one with the task that is closest to the root of the computation [20]. In addition, they propose a Set Based Victim selection *w*here each thief only attempts to steal from a subset of the other workers. We have implemented their methods in addition to other policies reported below.

V. EXPERIMENTAL SETUP

An exploratory software simulator (see Figure 2) is developed to rapidly assess the utility of different matching and scheduling procedures. The simulator is a time based simulator which uses two important “atomic” time units. The operating system atomic unit is referred to as a slice. A typical slice time is 1 – 30 milliseconds. Scheduling decisions are performed on a slice boundary. In addition, the simulator employs an atomic time unit referred to as a tick. System updates occur on a tick boundary. To mimic a realistic scenario we assume that a slice time is 20 milliseconds; we further assume that a tick represents 100 microseconds hence there are 200 ticks per slice. Other configurations have been used as well.

The simulator can be easily altered to evaluate a number of different configurations and task parameters. The overall system environment is equally flexible. Variables like the number of cores, power consumption per core, core frequencies, idle power consumption, slice time (in ticks), intra-core scheduling algorithms, stealing policies, and termination conditions can all be changed for individual experimental runs. Additional parameters include: 1) thresholds for the starvation/loaded status, 2) workload size, 3) task arrival rate, which in general follows a Poisson distribution, 4) task serving time, which in general follows an exponential distribution, and 5) task power consumption per tick, which is assumed to have a uniform distribution. These parameters have been selected based on discussion with experts from leading chip design companies.

The simulations are performed for all the formulated stealing policies. Each simulation is repeated several times with different random number generation seeds. Every simulation provides performance figures on a slice time basis for all the cores. Data is gathered on slice boundaries for each simulation of each stealing policy.

We report on two sets of experiments: Experiment 1: task scheduling with high initial rate of task generation, and Experiment 2: task scheduling with a steady arrival rate. The first scenario is typical of highly parallel loads, where in the first steps of computation many tasks are being generated. Initially, the system is saturated with new tasks, but after a while the system completes the processing of all the tasks in the current load. We refer to this scenario as the parallel workload scenario. The second mode is typical to communication and networking scenarios where tasks are generated at a more or less fixed rate and the system is usually in a steady state where the rate of processing tasks is about the same as the rate of task generation. This is referred to as the steady state scenario. Figure 3 illustrates the two scenarios via the size of the ready queue per slice time.

As noted, the work stealing procedure requires identifying a potential victim and selecting a task to be migrated. For the victim selection we have taken into account the amount of global knowledge, the set of potential victims, and the specific power performance goal. In terms of selecting the migrated task, it makes sense to assume that a victim core would like to volunteer its “worst” task as the task to be migrated. In this sense, we have identified HRRN as the most promising criteria for power agnostic work stealing. The victim is volunteering the task with the minimal HRRN for stealing. The HECN which takes into account EDP rather than expected execution time has been used as the basis for selecting the task to be migrated for the power-aware policies. In this case, the victim is volunteering the task with the minimum HECN as the task to be stolen.

A. *Stealling Policies, Legend and Avbbreviations.*

The following legend is used in the text and figures for the Power-aware (PAW) variants of the work stealing policies where HECN is used to determine the task to be volunteered by the victim core.

Random_MinHECN_Task; the stealer chooses a random core as a potential victim without knowledge of the core’s load. If that randomly chosen core is not loaded, then no stealing occurs. Otherwise, this victim core volunteers a task with the lowest HECN.

MaxLoaded_MinHECN_Task; the stealer identifies a loaded core with the largest ready queue as a victim. This victim core volunteers a task with the lowest HECN.

MaxMin_ HECN_Task; each loaded core (a potential victim) volunteers a task with lowest HECN. The stealer considers the tasks volunteered by all potential victims and finds a task with the highest HECN among all volunteered tasks. Hence the name MaxMin, implies that the MaxHECN task is selected from the available MinHECN tasks.

MaxRemainingService_MinHECN_Task; the service time of tasks remaining in the ready queue can be used to estimate the remaining core execution time and the energy that might be consumed. Therefore, the stealer picks the core with a ready queue that has the maximum remaining task service or execution time. The victim core volunteers a task with the lowest HECN.

MaxRemainingEnergy; the energy of tasks remaining in the ready queue indicates the energy that the core might consume. Hence, the stealer selects the core with a ready queue that has the maximum remaining task energy. In this case the victim has two options for volunteering tasks: 1) **MinHECN_Task;** the victim core volunteers a task with the lowest HECN. 2) **MaxEnergyTask;** the victim core volunteers a task with maximum energy.

MaxEnergyInLastKSlices; the stealer chooses a core that has consumed the maximum amount of energy in the last k slices of the simulation. Again, can choose between the: **MinHECN_Task** or the **MaxEnergyTask.**

MaxEnergyConsumed; the stealer opts for a core that has consumed the maximum energy so far in the simulation, and the victim has the same two options as in the previous policies: **MinHECN_Task or MaxEnergyTask**.

The PAG version of the above inter-core work stealing policies uses the HRRN ratio in place of the HECN to determine the task to volunteer.

VI. EXPERIMENTS AND RESULTS

This section reports the two types of experiments with work stealing conducted as part of this study and provides the results of these experiments.

A. *Experiment 1 - Multicore Task Scheduling for a Parallel Workload Scenario.*

In this set of experiments, a fixed workload simulation is performed in a system having a fast task arrival rate (parallel workload). These experiments are intended to study the behavior of the formulated policies and identify the policy that performs the best under this specific scenario. The four main performance figures provided from this experiment are the energy consumption variance, the average turnaround time, the peak ready-queue length, and the completion time of all the policies. The parallel workload scenario is depicted in Figure 3(a). According to the figure, the ready queue length is rapidly increasing in the first few time slices of the simulation and then gradually decreasing as the simulation progresses. Figure 4 shows the cores' energy consumption variance. This is used as an indicator of load balancing. It can be observed that, work stealing provides a reduction of about 18% in variance compared to PAG_NoSteal policy. The PAW_MaxMin_HECN_Task is the best stealing policy. The power-aware policies provide a marginally better power performance than the power agnostic method.

Figure 5 displays the turnaround time. In this case, the PAW_NoSteal policy has a lower turnaround time than PAG_NoSteal policy. This implies that power-aware intra-core task scheduling, without any stealing, lowers turnaround time by about 4%. By including stealing, the PAW_MaxMin_ HECN_Task is the best stealing policy and it improves (reduces) turnaround time further by approximately 31% compared to PAG_NoSteal policy. This shows that in the process of trying to gain power efficiency, the time factor is improved as well. This can be due to the fact that the EDP metric used in the selection criteria considers time along with power attributes. Again, power-aware is slightly better than power agnostic.

An experiment that measured the maximal size of the ready queues in each simulation of each stealing procedure has shown that the ready queues had reasonable sizes (up to 40 tasks per queue). Another experiment performed measured the task completion time. In this case, PAW_NoSteal policy increases the total completion by about 3.5%. This can be attributed to the fact that power-aware scheduling may increase task wait time and there is no stealing to help reduce wait time. On the other hand, stealing significantly reduces the completion time with PAW_MaxMin_ HECN_Task policy being the best stealer as it reduces the completion time by about 17%. Further experimental results are reported in [5].

From all of the results of this experiment, it can be seen that the PAW_MaxMin_ HECN_Task is the best stealing policy for a fast task arrival rate scenario. It significantly improves three important metrics, namely, energy consumption variance, turnaround time, and completion time.

B. *Experiment 2 - Multicore Task Scheduling for a Steady State Workload Scenario.*

For this test, a fixed workload simulation is performed in a system having a moderate task arrival rate. This emulates a steady state workload scenario as illustrated in Figure 3(b). In the first few time slices of the simulation, the ready queue length gradually increases. Then as the simulation progresses, the queue length remains steady for several slices thereby simulating a steady state workload scenario.

Figure 6 shows the cores' energy consumption variance. It is noticed that PAW_NoSteal policy performs slightly better than PAG_NoSteal policy by lowering the energy consumption variance by about 2%. By including stealing, the PAW_MaxEnergyInKSlices_MaxEnergyTask is seen as the best power-aware stealing policy. This policy further reduces the variance by 5% compared to PAG_NoSteal policy.

The PAG_MaxEnergyConsumed_MinHRRN work stealing policy provides a marginally better power performance than the PAW_MaxEnergyInKSlices_MaxEnergyTask method but it is not considered significant since it does not perform as well for the turnaround time metric discussed next.

Figure 7 displays the turnaround time. Again, PAW_NoSteal policy is better than PAG_NoSteal policy by almost 13%. The power-aware intra-core task scheduling coupled with inter-core work stealing further improves the turnaround time. The policy PAW_MaxEnergyInKSlices_MaxEnergyTask is again the best stealing policy with approximately 17% lower turnaround time compared to PAG_NoSteal policy. The power-aware policies are noticeably better than the power agnostic policies.

As in the case of parallel load, an experiment that measured the maximal size of the ready queues in each simulation of each stealing procedure has been performed. The result shows that the ready queues had reasonable sizes (up to 20 tasks per queue). Another experiment performed measured the task completion time. In this case, an important difference was noted compared to the previous experiment. The PAW_NoSteal policy is better than PAG_NoSteal policy with a 3% lower completion Furthermore, the best power-aware stealer of this experiment is again the PAW_MaxEnergyInKSlices_MaxEnergyTask policy with about 8% reduction in completion time compared to PAG_NoSteal policy.

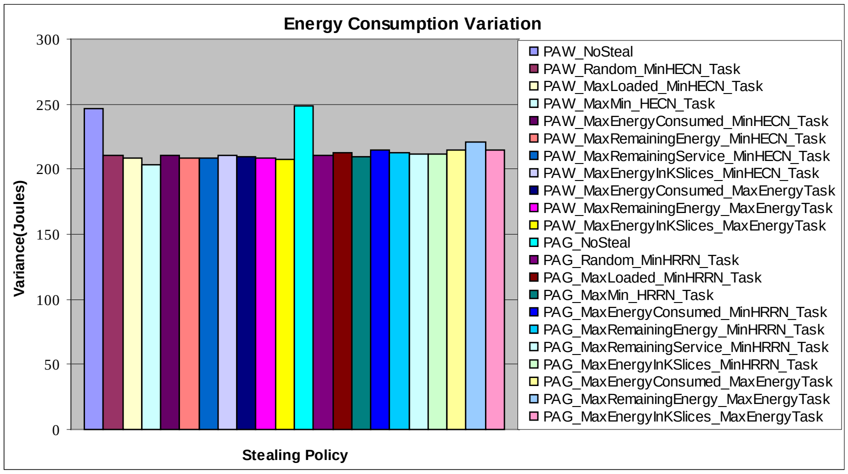


Figure 4. Energy Consumption Variance of the parallel load experiment

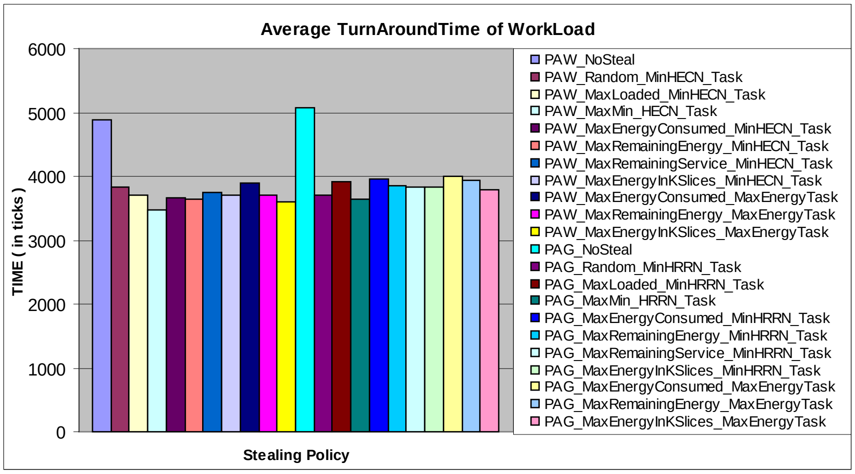


Figure 5. Average Turnaround time of the parallel load experiment

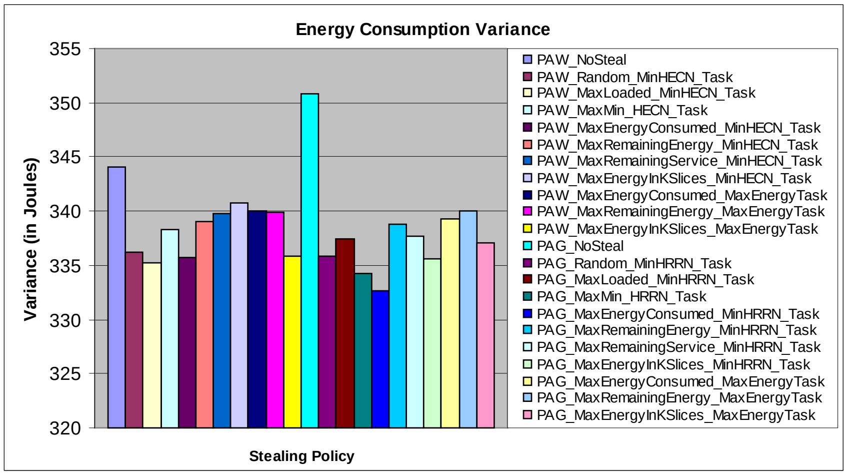


Figure 6. Energy Consumption Variance of the steady state load experiment

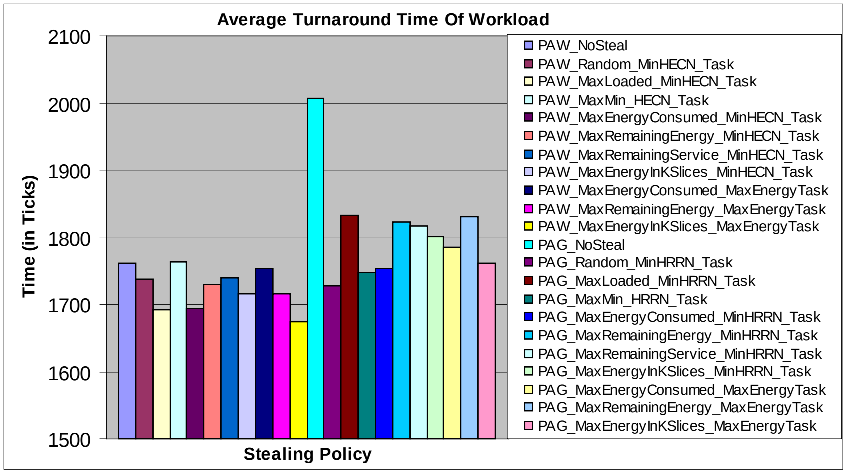


Figure 7. Average Turnaround time of the steady state load experiment

The PAG_MaxMin_HRRN_Task policy shows slightly better completion time compared to the PAW_MaxEnergyInKSlices_MaxEnergyTask policy, but it has a disadvantage since it fails to be the best in terms of efficiency in the energy consumption variance and turnaround time metrics.

VII. Result Evaluation

According to the data gathered, in every experiment, a power-aware policy emerges as the policy that successfully reduces energy consumption variance, turnaround time and completion time. In addition to accomplishing energy efficiency, the performance time has been improved as well. The PAW_MaxMin_ HECN_Task policy shows the highest potential with 18% reduction in energy consumption variance, 31% improvement in turnaround time, and 17% more efficiency in completion time compared to the PAG_NoSteal policy.

Furthermore, the key points noted from the combined results of all the experiments are as follows.

1. The PAW_MaxMin_ HECN_Task policy emerges as the best policy in Experiment 1. The reason for this might be because the MaxMin policy is the only policy that directly selects a task to steal by choosing the least power consuming task among the high power consuming tasks of all potential victim cores. All the other stealing policies, first select a potential victim core and then select a task from that chosen core. Therefore, a stealing policy that considers all the tasks in the system such as the MaxMin policy outperforms other policies.

2. PAW_MaxEnergyConsumedInKSlices_MaxEnergy_ Task policy is the most efficient policy in Experiment 2. This can be best explained by the following analysis. Excluding the MaxMin policy, all the stealing policies

first consider the power properties related to a core to determine a victim. Most properties are related to the number of tasks or type of tasks in the ready queue but only two of the policies consider the history of the core, namely, the MaxEnergyConsumedInKSlices policy which uses recent past data and the MaxEnergyConsumed policy which uses all the past data. Hence, a policy that considers properties related to the recent history of potential victim cores might have advantage over policies that ignore this information.

3. Based on the experiment results, the PAW_MaxMin_HECN_Task procedure is the policy with the overall most potential for power efficiency even if the task arrival rate is unknown. It is observed that this policy performs the best for cases with fast task arrival rate and also performs reasonably well in situations with steady task arrival rate.

4. In all the experiments, there is no significant difference in performance amongst many of the stealing policies. This could be attributed to the fact that the variations in work stealing are very minute and have subtle differences.

5. The turnaround time is improved much more than the power efficiency level in all the experiments. This implies that the EDP metric integrated into the HECN policy might be giving more consideration to the task time rather than to the task power.

VIII. CONCLUSIONS AND PROPOSALS FOR FURTHER RESEARCH

The primary goal of this research work is to develop efficient power-aware work stealing policies. In an attempt to achieve the desired goal, the following steps have been implemented. First, based on previous research [3][5], we have selected the HRRN as the benchmark for intra-core task scheduling and the derived HECN cost function that extends the HRRN policy to include power characteristics of tasks in the system.

Next, building on the new intra-core HECN policy, various inter-core work stealing policies have been explored. Several different power-aware variations of work stealing that consider power features of the cores and its tasks before identifying the task to steal have been formulated. Finally, an in-house exploratory simulator has been developed solely to evaluate the potential of the policies devised.

Extensive multicore experiments with work stealing policies have been performed. The outcome suggests that several power-aware policies have promising results where power efficiency is being attained along with a minimal effect on performance.

We plan to extend the reported research in several ways: first, we plan to examine the utility of additional heuristic evaluation functions. Next, we plan to consider affinity between tasks and cores (e.g., due to recent use of cache) in the stealing decisions. Additionally, we plan to connect our simulator to a vendor multicore board and use parameters of power and performance available at the hardware/firmware in the process of making scheduling decisions (as per the model depicted in Figure 1). This will enable fast and realistic exploratory simulations. Finally, we plan to incorporate DVFS policies and changing core states (e.g., shutting down cores) as a part of the scheduling decisions.

ACKNOWLEDGMENT

This research was partially supported by a grant from the Semiconductor Research Consortium. Additionally, this work was partially supported by the NSF grants: NSF CNS-1253292 and NSF CNS-1305302.

REFERENCES

[1] A. Merkel and F. Bellosa, “Balancing Power Consumption In Multiprocessor Systems,” in Proceedings of the 1st ACM SIGOPS/EuroSys European Conference on Computer Systems, 2006, pp. 403-414.

[2] A. K. Coskun, R. Strong, D. M. Tullsen, and T.S. Rosing, “Evaluating the Impact of Job Scheduling and Power Management on Processor Lifetime for Chip Multiprocessors,” in Proceedings Of The Eleventh International Joint Conference on Measurement and Modeling of Computer Systems, 2009, pp. 169-180.

[3] S. Shankar, D. E. Tamir, and A. Qasem, “Towards an OS-centric Framework for Energy-Efficient Scheduling of Parallel Workloads,” PPTDA-2013, the 2013 International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, July, 2013, pp. 21-28.

[4] C. Hyatt, G. LaKomski, C. Alvarado, R. Hay, A. Qasem, and D. E. Tamir, “Power-aware Task Matching and Migration in Heterogeneous Processing Environments,” the 2014 International Conference on Computational Science and Computational Intelligence, Las Vegas, US, 2014, pp. 3-8.

[5] S. Shankar, Power-aware Task Scheduling on Multicore Systems – Thesis Report, Texas State University, Computer Science, December, 2012.

[6] A. Silberschatz, P. B. Galvin, and G. Gagne, “CPU Scheduling,” in Operating System Concepts,8th ed.,John Wiley and Sons, 2008, pp.183-223.

[7] D. Tam, R. Azimi, and M. Stumm, “Thread Clustering: Sharing-Aware Scheduling On SMP-CMP-SMT Multiprocessors,” in Proceedings of the 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems, 2007, pp. 47-58.

[8] S. Boyd-Wickizer, M. F. Kaashoek, and R. Morris, “Reinventing Scheduling For Multicore Systems,” in Proceedings of the 12th Conference on Hot topics in Operating Systems, 2009, pp. 21-21.

[9] M. Rajagopalan, B. T. Lewis, and T. A. Anderson, “Thread Scheduling For Multicore Platforms,” in Proceedings of the 11th USENIX Workshop on Hot Topics in Operating Systems, 2007, pp. 35-44.

[10] J. Donald and M. Martonosi, “Techniques for Multicore Thermal Management: Classification and New Exploration,” in Proceedings of the 33rd International Symposium on Computer Architecture, 2006, pp.78-88.

[11] M. Kashif, T. Helmy, and E. El-Sebakhy, “A Priority-Based MLFQ Scheduler for CPU Power Saving,” in

Proceedings of the IEEE International Conference on Computer Systems and Applications, 2006, pp.130-134.

[12] K. H. Kim, R. Buyya, and J. Kim, "Power-aware Scheduling Of Bag-Oftasks Applications With Deadline Constraints On DVS-Enabled Clusters," in Proceedings of the Seventh IEEE International Symposium on Cluster Computing and the Grid, ser. CCGRID '07, 2007, pp. 541-548.

[13] G. Wu, Z. Xu, Q. Xia, J. Ren, and F. Xia, "Task Allocation and Migration Algorithm for Temperature-constrained Real-time Multicore Systems," in Proceedings of the IEEE International Conference on Cyber,Physical and Social computing, 2010, pp.189-196.

[14] X. Zhou, J. Yang, M. Chrobak, and Y. Zhang, "Performance-Aware Thermal Management via Task Scheduling," The Journal of ACM Transactions on Architecture and Code Optimization, vol. 7 issue 1, April. 2010, pp. 5:1-5:31.

[15] J. Quintin and F. Wagner, "Hierarchical Work-Stealing," in EuroPar'10 Proceedings of the 16th International Euro-Par Conference on Parallel Processing, 2010, pp. 217-229.

[16] Y. Guo, J. Zhao, V. Cave, and V. Sarkar, "SLAW: a Scalable Locality-aware Adaptive Work-stealing Scheduler," in Proceedings of the IEEE International Symposium on Parallel and Distributed Processing, 2010, pp.1-12.

[17] S. Agarwal, G.K. Mehta, and Y. Li, "Performance- based Scheduling with Work Stealing." Internet: http://www.cs.ucsb.edu/~gaurav_mehta/reports/cs290b.pdf, 2009 [retrieved: March, 2014].

[18] D. Sudarshan and D. Pooja, "LIBRA:Client Initiated Algorithm for Load Balancing Using Work Stealing Mechanism," in Proceedings of 2nd International Conference on Emerging Trends in Engineering and Technology, 2009, pp. 636-638.

[19] A. Robison, M, Voss, and A. Kukanov, "Optimization via Reflection on Work Stealing in TBB," in Proceedings of the IEEE International Symposium on Parallel and Distributed Processing, 2008, pp.1-8.

[20] K. Faxén and J. Ardelius, "Manycore Work Stealing,"in Proceedings of the 8th ACM International Conference on Computing Frontiers ACM, 2011, pp. 10:1-10:3.

Protecting and Sharing of Semantically-Enabled, User-Orientated Electronic Laboratory Notebook Focusing on a Case Study in the e-Science Domain

Tahir Farooq, Richard Kavanagh, Peter Dew
School of Computing, University of Leeds,
Leeds, United Kingdom
tahir_farooq@hotmail.com, scsrek@leeds.ac.uk

Zulkifly Mohd Zaki
Faculty of Science and Technology, Universiti Sains Islam
Malaysia, Nilai, Malaysia
zulkifly@usim.edu.my

Abstract— **We discuss the addition to an existing Electronic Laboratory Notebook (ELN) system, a means to permit the sharing of modelling data. One advantage is that sharing of such data is a means of assisting the publication process. This is done by presenting the modelling data and the reasoning behind its creation. This sharing of data is managed in a user sensitive fashion by restricting the release of data based upon the role someone performs. Further sensitivity is shown by fine-grained access control, which permits only part of the ELN to be shown. The performance of the solution presented is reviewed via quantitative analysis that showed a reasonable degree of end-user acceptance of the proposed approach.**

Keywords-electronic laboratory notebook; sharing; privacy; fine-grained access.

I. Introduction

In science and engineering, many international communities of researchers employ complex computational models. Such communities often use paper-based laboratory notebooks [1]. Research has previously focused on encouraging scientist in these communities to use an Electronic Laboratory Notebook (ELN) to create, store and retrieve provenance data about modelling, as a means of providing consistency of recording provenance data. The ELN was specifically designed to capture and store high quality metadata for the modelling process and modeller's reasoning, whereas previously this provenance metadata was recorded in an *ad-hoc* and unstructured fashion. This is in contrast to ELNs for physical experiments [2], where meta-data is often captured in a structured fashion.

In this paper, we advance on this previous work by allowing users to fully share electronic records that meet the technical and scientific requirements of such communities [3]. We refer to this is a community ELN called ELN-PS (protection and sharing of ELN) within a distributed, multisite research environment.

Working with one such community, namely the Atmospheric Chemistry Community we aim to enhance sharing of the modeller's data and its associated meta-data for the betterment of the community. This community studies aspects of chemical reaction mechanisms that take place in the lower atmosphere (troposphere). This community relies upon a highly comprehensive database of chemical mechanisms to drive their modelling process. This database is known as the Master Chemical Mechanism (MCM)[4]. It acts as the benchmark for this community and as such records in the database are carefully evaluated. The MCM database describes the detailed gas phase tropospheric degradation chemistry of a series of Volatile Organic Compounds (VOCs). Acting as the benchmark for the community it has a wide variety of atmospheric science and policy applications where detail knowledge of chemical reactions is required. MCMv3.2 [3], for example, contains 6,700 species involved in 17,000 reactions. Members of this community are involved in ensuring that the last research is evaluated and where necessary updates are made to the relevant MCM entries. One aspect of a community based ELN is to support the MCM updating process. If reviewers are given detailed information about the modelling that has been performed in the community then they are better able to understand the simulation results presented and the reasoning behind them. This therefore makes the updating of this central database easier.

Simulation data and its associated meta-data is an important commodity which may also be used by modellers for supporting publications, in that if their reasoning and process to obtain results can be followed by reviews the results and publication can be reviewed more readily. This process however requires careful management of the access to the associated data so that it respects the publication and evaluating processes. In this paper, the following contributions are made:

- An architecture that permits ELNs to be shared in a fashion that respects the publication process, allowing for the reviewing of ELNs and controlling the sharing of ELNs within the community. This includes a means to protect ELNs in an end-to-end fashion between the modeller and reviewer.
- A mechanism for the sharing of part of an ELN. This allows for a particular series of simulations known as a trail to be shared. This means only the data relevant to the evaluation and publication process is shared and not all the work of a given modeller.
- Finally, an assessment of an impact of sharing data within the selected community is discussed.

These advances extend the previous work on the ELN for individual modellers which is fully reported in the earlier paper [5] and is summarised here in Section 2. A previous user evaluation [5] of this work showed that it vastly improved the efficiency of the modelling process, promoted

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-339-1

good practice and facilitate easy and transparent knowledge transfer. The ELN for the community is expected like its predecessor to be relevant to other communities such as those that use detailed chemical reaction mechanisms such as GRI-Mechanism [6] and fields such as astrochemistry.

The rest of this paper is structured as follows: Section 3 details the requirements for the sharing of ELNs and in particular the lifecycle of the release of ELN data. In Section 4 the platform is introduced that provides the provenance sharing. This architecture is known as ELN-Protection and Sharing (ELN-PS). A web-based implementation of this architecture is discussed in Section 5, which is then used to elicite feedback from members of the atmospheric community in Section 6 with a qualitative analysis of the ELN-PS system. In the last section, we conclude and present our future work.

II. BACKGROUND

The work presented here extends our previous ELN for individual modellers [5]. The previous system is hence described here briefly in order to assist the understanding of this paper. The existing ELN is made up of three main components, namely: the Core ELN, the inline provenance node navigator (IPNav) and the notebook retrieval (see Figure 1). These components are described next:

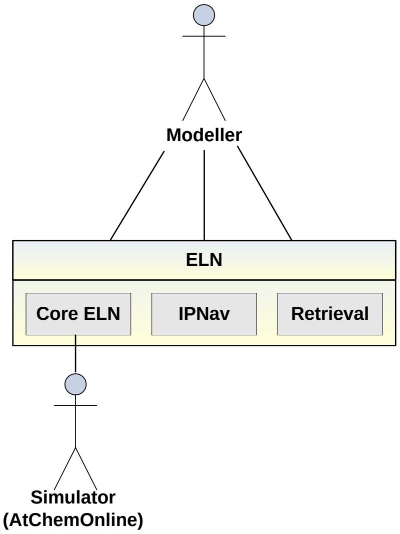


Figure 1. The ELN for Individual Modellers.

A. The Core ELN

This is principally responsible for executing simulation requests and recording all the parameters that go into the computational parametric modelling process. The modelling is performed by an external component called AtChem Online [7]. The core ELN records the output data from the AtChem modelling tool and links it to the provenance data, which indicates both the settings used as input and the user's original reasoning behind running the simulation. Simulations are performed iteratively and after each run the user is expected to change the parameters of the simulation, to further develop their model. The core records the step by step modelling process in a systematic and as far as possible, automated fashion. A feature of the core is that it supports the modeller in generating annotations to explain the rationale for making a parameter change at the time the change is introduced. Recording this reasoning at this point improves the quality of the annotations and their value to the modeller and other members of the scientific community once the notebook is shared. These annotations once made provide a narrative to the work of the modeller, giving complete coverage of their reasoning which includes both the successful and the unsuccessful formulations of the model. These annotations once combined with details of the modelling process provide the meta-data which we call the inline provenance of the model.

B. Inline Provenance Node Navigator (IPNav)

The IPNav [3] structures and displays to the end user the provenance as a graph/tree structure. It thus fully represents the inline provenance gathered as part of a series of successive interations of the model. It allows this provenance to be navigated and presents the modeller with the ability to compare different iterations of the model's development, using an inbuilt differencing tool. This viewer is particularly important for third party users of the ELN, whom of course did not develop the model and hence were not privy to the decisions and reasoning process of the original modeller.

C. Retrieval

The ELN retrieval function provides the ability to search and recover from the database past models which then allows the inline provenance node navigator to display the individual runs of the ELN. Its also allows the user to view both the experimental data and its provenance.

In addition to the advancements made with sharing, it should be noted since the previously reported version of the ELN was produced, an evaluation study identified that there was a need to reduce the time and complexity of setting up the ELN on a modeller's local computer. This was because it requires a number of third party software, namely: Python, Python Yet Another Markup Language (PyYAML), Natural Language Toolkit (NLTK), My Structured Query Language (MySQL), curl, diff, NetBeans and Java's Software Development Kit (SDK). This issue was resolved by using virtualisation which provided a prefabricated environment for the ELN.

III. REQUIREMENTS FOR ELN-PS

In Section 2, the ELN for individuals was discussed, including that of the generation of provenance meta-data. This provenance data that is generated assists the publication process and is a valuable resource for e-Scientists as it helps with: the repeatability of experiments, tracking experimental runs, managing the data generated, verifying experiment results and acts as a source of experimental insight [8]. The lifecycle and associated requirements that govern the release of this provenance data are now to be discussed. The ELN lifecycle process is concerned with the management of the end-to-end provenance flow from the initial models creation to the use in the wider ELN community. This lifecycle, as shown in Figure 2, highlights the ELN protection and sharing requirements.

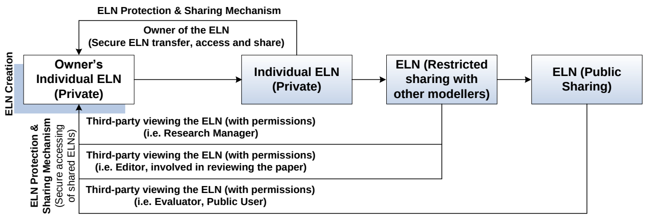

ELN Protection & Sharing Mechanism
Owner of the ELN
(Secure ELN transfer, access and share)
ELN Creation
Owner's Individual ELN (Private)
Individual ELN (Private)
ELN (Restricted sharing with other modellers)
ELN (Public Sharing)
ELN Protection & Sharing Mechanism
(Secure accessing of shared ELNs)
Third-party viewing the ELN (with permissions)
(i.e. Research Manager)
Third-party viewing the ELN (with permissions)
(i.e. Editor, involved in reviewing the paper)
Third-party viewing the ELN (with permissions)
(i.e. Evaluator, Public User)

Figure 2. ELN Lifecycle Process.

Requirement 1: The principle of the ELN protection and sharing control is that the owner of the ELN (modeller) is required to share the personal ELN to the wider ELN community in a secure way. The protection and sharing of ELN thus has to be followed in a staged process. There is therefore three stages of release of an ELN in the wider ELN community: 1) “Private” so that only ELN owner has access 2) “Shared” enables ELN owner (modeller) to share personal ELN with other modellers 3) “Public”, so that any community member (if ELN owner has allowed) can viewan ELN.

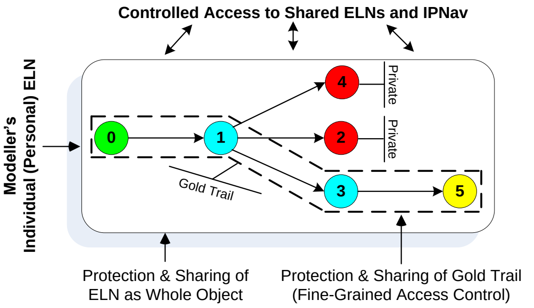

Controlled Access to Shared ELNs and IPNav
Modeller's Individual (Personal) ELN
0
1
2
3
4
5
Private
Private
Gold Trail
Protection & Sharing of ELN as Whole Object
Protection & Sharing of Gold Trail (Fine-Grained Access Control)

Figure 3. Fine-grained Access Control of ELN Trails.

Requirement 2: At some stage in the modelling process, community members may be interested in sharing only part of their personal ELN, i.e., access control to an ELN at a fine-grained is required. This is explored in Figure 3, where each simulation run is coded by colour indicating its position in the trail. For example green indicates the base run, red the dead ends and yellow highlights the gold/latest simulation run. Cyan means, the intermediate runs.

Fine-grained access control, is the application of protection and sharing rules to control access to parts of an ELN's provenance trails. This ensures modellers have the flexibility to share certain parts of their ELN trails with others in the community. Therefore, by default, every navigation node is tagged as “private” and the modeller has the choice of applying access control permissions from a pool of accessibility options. One such option is to share the trail, with the node that represents the best experimental case as the final node, this is known simply as the “gold trail”.

Requirement 3: During the release of an ELN to the community it will be required for many different people to be able to access the data. These people will have different roles, such as a researcher's supervisor, or a reviewer. It will be required to moderate the access to a given ELN based upon these roles.

Evaluation of Requirements: To assess the meeting of these requirements a qualitative user-orientated evaluation to assess the value of the ELN protection and sharing mechanism will be performed (see Section 6).

A. *Scenario Cases*

The requirements are drawn from the following scenario cases, which are derived from the working practices of the atmospheric chemistry project EUROCHAMP-2 [3], though remain generalisable to other communities with similar requirements. The scenario cases are divided into three main parts: a) the sharing of a whole ELN; b) sharing of ELN provenance trails at a fine-grained level; and c) management of ELNs in the central repository. These cases highlight the relevant characteristics and working procedures of modellers sharing ELNs.

1) *Part-1:Sharing of a whole ELN*

Helen is a modeller working in her local laboratory. After finishing simulating a toluene chamber experiment on her local computer, she transfers the first version of ELN (H-v1) into the community repository using ELN-PS system. During the transfer process, the default access of the ELN is set to private and its owner Helen. Hence, the ELN is neither visible or accessible to anyone other than Helen.

Helen shares the first version of ELN (H-v1) with her research manager Peter to get feedback. Helen allows Peter to access all the trails of ELN (H-v1), i.e., the whole ELN. Peter as a research manager examines all simulation runs of the ELN and suggest some updates in run 5 of the simulation. Helen takes his advice and transfers the second version of ELN (H-v2) into the repository and shares it with Peter.

2) *Part-2: Sharing of ELN provenance trails at a fine-grained level*

After examining the final ELN, Peter advises Helen to allow Mark to access gold trail of the ELN (H-v2) for review purposes as part of publishing a paper. Mark acting as an editor examines the ELN gold trail of the toluene chamber experiment. The results in the latest/gold trail helps him to make a positive recommendation to publish the paper.

After publishing the modeller's results, Helen marks the gold trail of ELN (H-v2) as public thus making it available for other community members.

ISBN: 978-1-61208-339-1

3) Part-3: Management of ELNs in the central repository

Jill, another researcher working on toluene experiment recently joined the research group. She searches through the ELN-PS system to find related ELNs in the community repository. The search retrieves the shared ELN(s). For the previous retrieved ELN only the gold trail is displayed, because as a public user she is restricted to only viewing the published trail.

Helen now leaves the research group and her user status is blocked by Lindsey who is acting as systems manager. During the routine management searchers on the ELN repository, Andrew as a data manager finds that the owner of the toluene ELNs has left the group. Andrew follows the research group policy and allows Jill to access all toluene ELNs created by Helen thus allowing her to proceed with her research.

IV. ELN-PS SYSTEM ARCHITECTURE

An overview of ELN-PS system architecture is shown in Figure 4. A modeller with their own individual ELN is given the option to transfer it to the community ELN repository, via the use of an ELN transfer protocol. The provenance information for a particular simulation and its runs is transferred as Resource Description Framework (RDF) [1] metadata. These metadata contain the process provenance and associated annotations for simulation runs.

A modeller at the start of the transfer process uses the simulation retrieval function in the ELN for individuals. Once a simulation is chosen, the ELN performs an export of the simulation data and provenance. The “Transfer ELN” function of the ELN-PS system then allows the modeller to import the selected simulation and its associated runs into the community ELN repository. In reverse order, the “Download ELN” function of the ELN-PS system allows the modeller to download individual ELN from the community ELN repository.

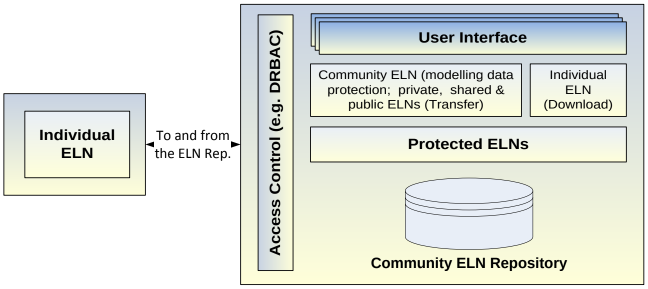

Individual ELN
To and from the ELN Rep.
Access Control (e.g. DRBAC)
User Interface
Community ELN (modelling data protection; private, shared & public ELNs (Transfer)
Individual ELN (Download)
Protected ELNs
Community ELN Repository

Figure 4. ELN-PS System Architecture.

The access control layer in the ELN-PS system is built on the Dynamic Role Based Access Control (DRBAC) framework. DRBAC provides authorisation to the community ELN repository based on the assigned roles of users. The reason for using role based access control is that, it gives a clear understanding of responsibilities to each user. The roles are defined according to the job competency, authority and responsibility within the organisations to regulate access to the ELNs. In eScience communities like EUROCHAMP-2, the roles change dynamically (e.g., a person may at different time perform the role of a research manager and at other times the modeller role). Further in the wider ELN community, the research laboratories may need to define a custom description in the ELN access control mechanism. It is therefore important to dynamically allocate roles to the users and dynamically allocate permissions to the roles. Roles are used to embody the authority and responsibility of the main actors of the community in the system. The responsibilities of such roles therefore guide the need for access to the ELNs secured in the repository. The role based access control in ELN-PS is based upon the National Institute of Standards and Technology (NIST) Role Based Access Control (RBAC) model [9]. Further details on DRBAC can be found in the related work section of the paper. The role hierarchy for ELN-PS system is shown in Figure 5. These roles and their associated permissions are assigned dynamically to the community members as required so that they may perform different tasks such as: transfer, share and view ELNs.

The ELN-PS system role hierarchy is organised into three categories:

i) *Group*: Member of the wider ELN Community which has three sub-roles namely; Modeller, Research Manager and Editor;

ii) *Public*: Member of Public Community Group which has three roles namely Evaluator, Public User and Public Blogger;

iii) *Admin*: Administrators; has two roles namely; System Manager and Data Manager.

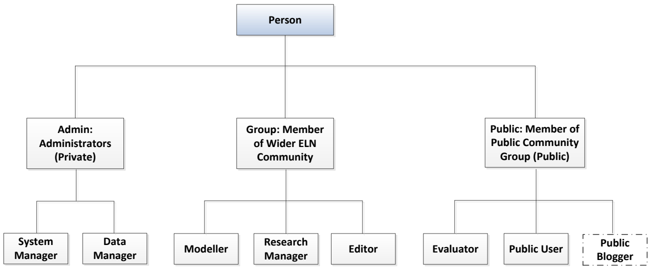

Person
Admin: Administrators (Private)
Group: Member of Wider ELN Community
Public: Member of Public Community Group (Public)
System Manager
Data Manager
Modeller
Research Manager
Editor
Evaluator
Public User
Public Blogger

Figure 5. Role Hierarchy.

The modeller role, as shown in Figure 6, is further divided into three sub roles each of which deals with a different stage of release of the ELN data (see requirement 1):

i) *Modeller-Private role* can: a) transfer personal ELNs into the central repository; b) share the whole ELN; c) retrieve and view personal ELNs; d) download personal ELNs; and e) view/add comments.

ii) *Modeller-Public role* give permissions to the modeller to share and view gold/latest simulation trail of the ELN.

iii) *Modeller-Selective role* allows the ELN owner to share any simulation trail of the ELN with other modellers.

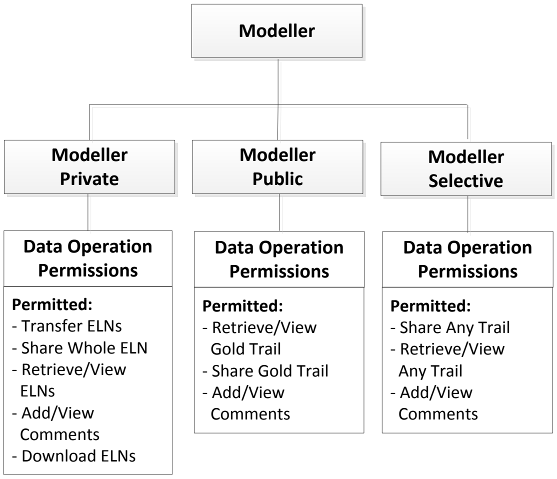


Figure 6. Modeller Role Properties.

The remaining roles are as follows: Research Manager allows a supervisor to view assigned researchers (modellers), view their shared ELNs and view/add comments. The Editor role allows the review process of a paper. It allows an editor to view the shared gold/latest trail and discuss it with fellow editors confidentially. The Public User role: can only view the final (publish) gold trail, provided it is shared by the ELN owner. The System and Data manager roles are associated with the management of the system including: archiving old ELNs, user management, roles management, role assignment, and role activation/de-activation.

The ELNs are transferred, shared and accessed through respective user interfaces of the ELN-PS system. The person-role and role-permission sessions are created dynamically within the system to open the static bindings of three main components of the traditional RBAC system: persons (users), roles and permissions. The access control layer in the ELN-PS system mainly addresses the authorisation process, which is based on the mapping of roles and permissions to ensure the person access to different services and functions. The user identification is done separately using a Form-based identification process [10]. The identification certifies the person credentials for the ELN-PS system. Figure 7 represents the internal view of person authorisation flow of the underlying security architecture. This is divided into two parts:

i) DRBAC Module
This module provides the allocation of community roles and associated permissions at run time for the entire session.

ii) ELN Access Control Module
The ELN access control functions and procedures to perform actions like Transfer ELNs, Share ELNs, View ELNs etc, which are provided in this module.

A person is authorised to access the community ELN repository according to the specific assigned roles. The authorisation process works with the generation of unique authorisation keys and security code for every person at run time. After the identification, when the access control request is received from the access control layer, the internal process of authorisation is started.

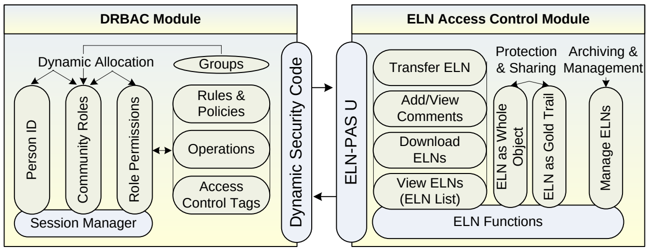


Figure 7. Authorisation Process in ELN-PS System.

Person, roles, permission and related identification keys are stored in the DRBAC database. If a person request authorisation two keys are generated. The first key contains the person and role identifications and the second key contains role and permission identifications. If a person is allocated multiple roles then multiple pairs of keys are generated for that particular person. The same mechanism is used when a role is allocated multiple permissions, i.e., the multiple keys are generated for that particular role. After successful processing of the unique authorisation keys, the secured information is forwarded for generation of a unique dynamic security code for every user. This security code is then combined with the “ELN-PAS U” (ELN Protection, Access and Sharing Unit) to access the ELN metadata. ELN-PAS U carries out the following operations:

i) Transfer ELNs: This allows modeller to select local ELNs using import function and transfer into the central repository.

ii) Share ELNs: Modeller can share personal ELNs as a whole or selected provenance trails with other community members like research manager, editor, evaluator etc.

iii) Access ELNs: This contains three types of access levels: a) “Private” so that only ELN owners have access on their personal ELNs; b) “Shared” enables modellers to share ELNs with other modellers and allows to view shared ELNs; and c) lastly “Public”, so anybody can view public ELNs within or across the community.

iv) View/Add comments: This allows modeller to exchange comments privately with research manager or editor on a particular ELN or its provenance trail.

v) Archive and manage ELNs: This is for the data manager to archive and manage the: a) old ELNs; and b) ELNs of the modeller who left the research group.

The authorisation process in the ELN-PS system ensures that:

i) Roles are allocated dynamically (at run time). If a role is not assigned to a person, then the related authorisation key will not be added in the security code. So, a person cannot use that role. The security code stops working if a role is de-activated or blocked at any moment during program execution.

ii) Permissions are allocated dynamically (at run time) to the roles and could be activated or de-activated at any moment of the processing time.

iii) With the use of the dynamic security code, ELN-PAS unit works on a safe and protected mechanism. It prevents a private ELN becoming available automatically to any person in the community without the proper permissions being given by the modeller/data owner.

V. Web Based ELN-PS System

In this section, we introduce the implementation of ELN-PS system that was created for the purpose of eliciting feedback from members of the EUROCHAMP-2 community. The architecture is presented as a Web based implementation and is shown in Figure 8.

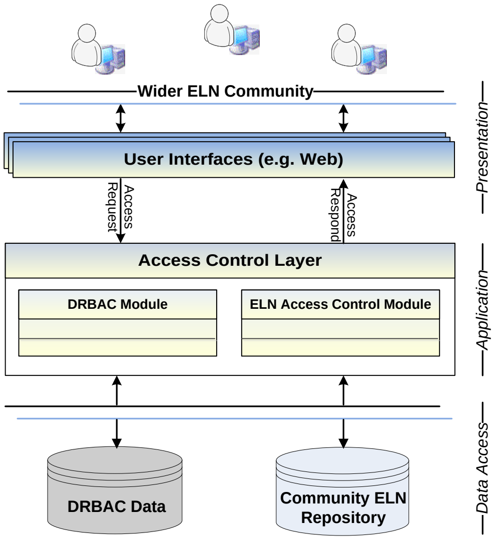
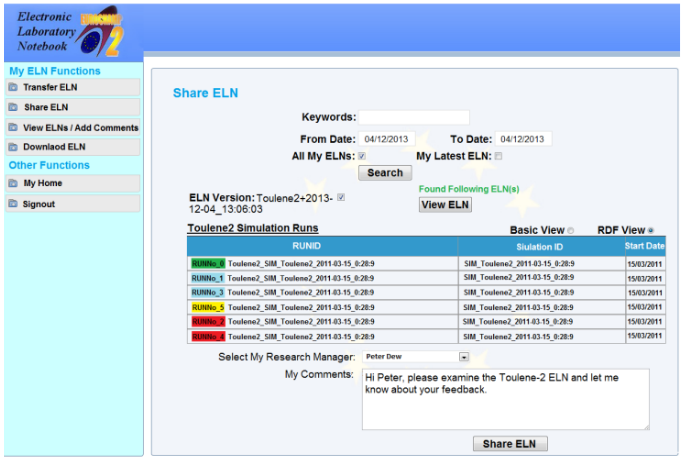


Figure 8. Implementation of ELN-PS System.

The authorisation process in the ELN-PS system was discussed previously in Section 4 so will not be repeated here. The implementation is based on 3-tier web architecture. It is coded in PHP, JavaScript and HTML and is hence reliant on a web browser to render the application executable [11]. The advantage of using a web based implementation is that it copes well with the distributed nature of the community in question. It presents the ability to update and maintain the web application without distributing and installing software on many different user computers. The geographically distributed environment and nature of users, requires the use of a centralised protection and sharing system that is accessible anywhere and is platform independent.

MySQL was used to implement the backend database. For the server-side scripting, PHP was used along with the semantic library for PHP ARC2 [12] in order to read the RDF data, associated with each ELN. A key aspect of the development was the Graphical User Interface (GUI), as even if the required functionality was met, if the GUI was hard to understand or unfriendly, then the program will ultimately be a failure. Interface design encompasses three distinct, but related constructs: usability, visualisation, and functionality [13]. A fourth component of accessibility has emerged as a critical factor in regards to the design of Web-based applications. The ELN-PS system thus uses a Cascading Style Sheet (CSS) for styling information. An example, rendering of the ELN-PS GUI showing the "Share ELN" interface is shown in Figure 9.



Share ELN

Keywords:

From Date: 04/12/2013 To Date: 04/12/2013

All My ELNs: ☑ My Latest ELN: □

Found Following ELN(s)

ELN Version: Toulene2+2013-12-04_13:06:03 ☑

Toulene2 Simulation Runs

Basic View ○ RDF View ●

RUNID		Siulation ID	Start Date
RUNNo_0	Toulene2_SIM_Toulene2_2011-03-15_0:28:9	SIM_Toulene2_2011-03-15_0:28:9	15/03/2011
RUNNo_1	Toulene2_SIM_Toulene2_2011-03-15_0:28:9	SIM_Toulene2_2011-03-15_0:28:9	15/03/2011
RUNNo_3	Toulene2_SIM_Toulene2_2011-03-15_0:28:9	SIM_Toulene2_2011-03-15_0:28:9	15/03/2011
RUNNo_5	Toulene2_SIM_Toulene2_2011-03-15_0:28:9	SIM_Toulene2_2011-03-15_0:28:9	15/03/2011
RUNNo_2	Toulene2_SIM_Toulene2_2011-03-15_0:28:9	SIM_Toulene2_2011-03-15_0:28:9	15/03/2011
RUNNo_4	Toulene2_SIM_Toulene2_2011-03-15_0:28:9	SIM_Toulene2_2011-03-15_0:28:9	15/03/2011

Select My Research Manager: Peter Dew

My Comments: Hi Peter, please examine the Toulene-2 ELN and let me know about your feedback.

Figure 9. Share ELN Interface.

The "Share ELN" function allows a modeller to share individual ELNs as a whole object with other community members such as with their research manager. The "Add Comments" section allows modeller to exchange comments privately with their research manager or an editor. These comments are then recorded against the ELN as part of the life cycle information, these comment then may be retrieved at a later date.

VI. Qualitative Evaluation with End Users

In this section, we perform a qualitative evaluation of the ELN-PS system. Qualitative evaluation captures descriptive data collected through the observations and interviews with end users and gives a voice to the participant's experiences [14]. It is used here as a mechanism for assessment on how well the ELN-PS performed. In this research, the goal of conducting qualitative evaluation with the end users was to determine the potential value, likely advantages and disadvantage of using:

i) The ELN-PS system;

ii) An DRBAC mechanism to protect and share ELNs;
iii) The concept of fine-grained access control to share only the selected ELN trails such as the gold simulation trail.

The evaluation plan included:
i) An introduction to the protection and sharing of ELN followed by questions/answers session;
ii) The demonstration of each scenario case in the ELN-system; and
iii) The collection of end user's feedback using a specific evaluation questionnaires, designed for this purpose.

A likert scale was used to assess the answer of each question in the evaluation process. Values were ranked from 1-5, 1 being very poor, 2 = poor, 3 = good, 4 = very good and 5 = excellent. After the demonstration of each scenario case, users provided feedback. The answers obtained from two members of the community are given in Table 1.

TABLE 1. ANSWERS FROM USER SURVEY.

Questions To Users	User	
	1	2
i) Do you understand the ELN protection and sharing process in the ELN-PS system?	3	4
ii) Do you value the DRBAC mechanism, adopted to protect and share the ELN?	4	3
iii) Do you think the role names, defined in the ELN-PS system give clear understanding to the people about their position?	3	4
iv) Are you satisfied with the privacy policy to protect and share ELN as whole object?	3	4
v) Do you understand the concept of fine-grained access control to share only a selected ELN trail (like gold simulation trail)?	3	4
vi) Do you see the value in giving an option to the ELN owner to restrict a third-party to view just the gold simulation trial?	4	4
vii) Do you think, it is good to provide extra functionality to share ELN trails other than gold trail?	4	3
viii) How you rate the design of the user interfaces? Is it clear and user friendly?	3	3

At the end of the evaluation, the recommendations and comments from the participants was recorded. A sample of their recommendations and comments are provided below:

A. Recommendations:

User 1: "Some adjustments to the design of the interface to improve usability".

User 2: "What I can see is that: a) the system should send an email auto notification to the person (e.g., supervisor) who will share the data with modeller, b) provides a variety of searching tips (e.g., search by date, by name of the simulation, search by range of date, search by EUROCHAMP chambers or search by collaborator partners), and c) includes the simulation trails (i.e., IPNav), so that the modeller can easily visualise the trails of the simulation runs".

B. Comments:

User 1:
1. "Confidence is the key to use".
2. "Facilitates remote supervision of student(s) / scientist(s) – gives the ELN unique educational aspects".
3. "Fine-grained access control is important as some users will want to share more/less of the ELN than others – again a key aspect of its usability".
4. "This is a key aspect if the community is going to really use this system as a primary scientific tool".
5. "Option to even "delete" ELN will make people feel safer".

User 2:
1. "Currently, I think the role names are sufficient unless if they changes from the users".
2. "This is a very good idea of protecting the provenance data".
3. "The user interfaces need to be improved".

The results were overall very encouraging, both participants rated the value of this protection and sharing mechanism between good and very good (i.e. 3 or 4 out of 5). They saw the value of being able to securely share the ELN as whole object and partly (i.e. at fine-grained level) with third parties (like research managers, editors, evaluators, etc.). These results can, therefore, be considered as an initial feedback before going into the larger community for further evaluation. The major concerns they have shown was the trust relationship with the third party (i.e. the data centre where the ELNs will be kept). User 1's comments in regards to confidence relates to the trust in the third party storing the data. We have presented a role-based access model surrounding the storage of ELNs but security must encompass the system as a whole and the end users need assurances that the service provider will maintain the relevant security around the ELNs stored.

VII. RELATED WORK

Access control is critical to information security and data protection. Within the Atmospheric Chemistry community, sharing of digital resources with a different degree of sensitivity is crucial as it ensures modelers are confident with the protection of their data. Details of the comparison of various access control models have been discussed in [15][16]. Based on the enhanced dynamic role based access control, this paper has introduced the ELN protection and sharing mechanism for secure access and sharing of ELNs from a central repository as whole objects or its elements (provenance trails) at fine grained level.

Generally, roles are defined as either static or dynamic. Static roles are normally based on a strictly defined association of users to the system that are established early and rarely change, dynamic roles ensures these associations are assessed at runtime as requests for access are made [17]. The NIST model [9] discusses RBAC and provides: a strict definition of RBAC sets and relations, while also defining a common vocabulary, setting the scope of the RBAC uniform features and introduces a functional specification providing administrative, review and system functions. It however does not incorporate a scalability attribute or permissions which deny access (i.e., negative permissions).

PERMIS [18] is a RBAC authorisation system that uses X.509 attribute certificates [19] to hold user's roles. Authorisation decisions are made through the PERMIS's access control decision engine based on the roles assigned to the user. PERMIS however does not define a mechanism for aggregating attributes from multiple authorities, where the user is known by different names at each authority. In AAA (Authentication, Authorisation and Auditing/accounting) [20], the RBAC framework is based on PERMIS which uses federated identity providers. Roles are used to identify users only to provide static access. The rest of the security is applied through the use of public and private keys. The concept of DRBAC is also not discussed in relation to sharing experiment metadata as is the case here.

In the eScience domain, CARMEN [21] and myExperiment [22] also discuss the data protection and sharing issues in a distributed environment. However, dynamic access control for sharing of metadata is limited or not discussed. Access control at a fine-grained level and varying descriptions of roles among different research groups is also not addressed.

ROWLBAC [23] explores the relationship between OWL and RBAC. It proposed two different approaches in the representation of roles i.e. roles as classes and roles as values; using a standard description logic reasoner. The role permissions defined are however limited. For example it explains the permissions on the basis of a Boolean function and does not discuss the access control at a fine grain level.

Smirnov et al. [24] present a RBAC model that is extended by adding a trust factor for a distributed environment. In this work, it gives every trust value for each user by introducing trust management to access control. It does not however, discuss the dynamic access control at fine grain level regarding metadata such as an ELN.

In [25], a model of the context-based access control for the information shared in a smart space is proposed. It uses open source Smart-M3 platform [25] and is built on the combination of the role based and attribute based access control models. Roles are assigned dynamically based on the participant’s trust level. However, in this research, the co-ordination of roles in a hierarchy model and activation/de-activation of multiple roles are not discussed. Further the access control about the flow of data among different roles is not addressed. For example, like in the Atmospheric Chemistry domain, the experiment metadata is not accessible for Public role unless it is evaluated.

Carminati et al. [26][27] propose an access control system based on the Semantic Web technologies for social networks. It enables granting of access based on ‘friendship” relation with the resource owner and on evaluation of the confidence level of the user. This works seems suited with the Atmospheric Chemistry community where modeller collaborated with other modeller in a laboratory or between other laboratories. However, in this research, we take it one step further providing access and share simulations metadata at a fine grained level.

VIII. CONCLUSIONS AND FUTURE WORK

The aim of this research was to study how to share modelling data and its provenance across a research community. The proposed architecture has been realised and in this instance tailored to the EUROCHAMP-2 community. It demonstrates the sharing of ELNs in a secure manner. It further shows how the DRBAC model allows for the protection of ELN provenance trails at a fine-grained level, thus ensuring that only data relevant to the evaluation and publication process is shared.

The qualitative evaluation demonstrated how a role based access system could be understood and accepted by a research community. In addition it showed how offering user’s fine-grained access control over what they share elicits acceptance, especially when a community is sensitive to the sharing of a trail of important simulation runs.

This research can be considered to be an initial step in defining an access control model to protect and share ELNs within research communities. Future work will be to introduce the ELN-PS system to other eScience communities. The changes required are considered to only be need in the ELN system for individuals, so it can be tailored to a given community, leaving the RBAC system intact. In order to get more conclusive results on the value of the ELN-PS system, a larger set of ELN modellers is needed. However, before going into the larger community for further evaluation, the issue of establishing a trust relationship between end-users and service providers will need addressing. Only when a credible service provider such as the British Atmospheric Data Centre (BADC) [28] in EUROCHAMP-2 case, with robust plans for the safe storage of ELN data, will a community be willing to share their ELNs. Integration of technologies, such as Secure Socket Layer (SSL) Protocol [29] and encryption/decryption algorithms [30] into the ELN-PS system are also required to instill greater confidence from end-users.

Further, we aim to define the ELN as a service in the cloud. Cloud computing delivers the infrastructure, platform and software as services, which are made available by subscription in a pay per use model [31]. By defining the ELN service in the cloud, it could be managed on an on demand basis. Cloud computing would also offer a highly scalable solution which would be able to meet the ongoing demands of several different ELN oriented research communities.

REFERENCES

[1] C. Martin, M. Haji, P. M. Dew, M. J. Pilling, and P. K. Jimack, “Semantically-enhanced model-experiment-evaluation processes (SeMEEPs) within the atmospheric chemistry community,” Second International Provenance and Annotation Workshop, IPAW, Springer-Verlag, 2008, pp. 293-308.

[2] Taylor K., Essex J.W., Frey J.G., Mills H.R., Hughes G., and Zaluska E., “The semantic grid and chemistry: experiences with CombeChem,” J. Web Semantics, 2006, 4 (2), 84-101, doi:10.1016/j.websem.2006.03.003.

[3] Zulkifly Mohd Zaki, Peter Dew, Mohammed H. Haji, Lydia MS Lau, Andrew Rickard, and Jennifer Young, “A user-orientated electronic laboratory notebook for retrieval and extraction of provenance information for EUROCHAMP-2,” The 7th IEEE International Conference on e-Science, December 2011. pp. 371-378.

[4] Jenkin, M.E., S.M. Saunders, V. Wagner, and M.J. Pilling, “Protocol for the development of the master chemical mechanism, MCM v3 (Part B): tropospheric degradation of aromatic volatile organic compounds,” Atmos. Chem. Phys., 2003, 181-193, doi:10.5194/acp-3-181-2003.

[5] Mohd Zaki Z, Dew PM, Lau LMS, Rickard AR, Young JC, Farooq T, et al., “Architecture design of a user-orientated electronic laboratory notebook: A case study within an atmospheric chemistry community,” Future Generation Computer Systems, 2013, vol. 29, issue 8, pages 2182–2196.

[6] M. Frenklach, T. Bowman, and G. Smith, (n.d.), “GRI-mechanism,” http://www.me.berkeley.edu/gri-mech/, [retrieved: 11th April, 2014].

[7] The University of Leeds. (n.d.), “AtChemOnline homepage,” https://atchem.leeds.ac.uk/webapp/, [retrieved: 11th April, 2014].

[8] Greenwood, M., Goble, C.A., Stevens, R.D., Zhao, J., Addis, M, Marvin, et al., “Provenance of e-Science experiments experience from bioinformatics,” The UK OST e-Science second All Hands Meeting (AHM'03), 2003, pp. 223-226.

[9] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramoul, “Proposed NIST standard for role-based access control,” ACM Transactions on Information and System Security, 2001, vol. 4, Issue 3, pages 224-274.

[10] Oracle. (2010). “The Java EE 5 tutorial,” http://docs.oracle.com/javaee/5/tutorial/doc/bncbe.html, [retrieved: 11th April, 2014].

[11] M. Miller, “Cloud computing: web-based applications that change the way you work and collaborate online,” A Book by Que Publishing, 2009.

[12] Annon, “Easy RDF and SPARQL for LAMP systems,” https://github.com/semsol/arc2/wiki, [retrieved: 11th April, 2014].

[13] Information Resource Management Association, “Instructional Design: Concepts, Methodologies, Tools and Applications,” IGI Global, 2011, DOI: 10.4018/978-1-60960-503-2.

[14] Breier, J. and Hudec, L., “New approach in information system security evaluation,” IEEE first AESS European Conference on Satellite Telecommunications (ESTEL), 2012, pages 1-6, IEEE, DOI: 10.1109/ESTEL.2012.6400145.

[15] S. M. Hasani and N. Modiri., “Criteria specifications for the comparison and evaluation of access control models,” International Journal of Computer Network and Information Security(IJCNIS), 2013, vol. 5, pp. 19-29.

[16] P. N. Mahalle, B. Anggorojati, N. R. Prasad, and R. Prasad, “Identity authentication and capability based access control (IACAC) for the internet of things,” Journal of Cyber Security and Mobility, 2013, no. 4, pp. 309-348.

[17] J. Odell, H.V.D. Parunak, S. Brueckner, and J. Sauter, “Changing roles: dynamic role assignment,” Journal of Object Technology, ETH Zurich, 2003, pp. 77–86

[18] Sacha Brostoff, M. Angela Sasse, David Chadwick, James Cunningham, Uche Mbanaso, and Sassa Otenko, “R-What? development of a role-based access control (RBAC) policy-writing tool for e-Scientists,” Software: Practice and Experience, 2005, pp. 835-856.

[19] International Telecommunication Union (ITU), “Information technology - open systems interconnection - The Directory: Public key and attributes certificate frameworks,” ITU-T Recommendations X.509, March 2000.

[20] R. O. Sinnott, A. J. Stell, and J. Watt, “Advanced security infrastructures for grid education,” 10th World Multi-conference on Systemics, Cybernetics and Informatics, (WMSCI 2006), 2006, pages 182-196.

[21] Martyn Fletcher, Bojian Liang, Leslie Smith, Alastair Knowles, Tom Jackson, Mark Jessop, et al., “Neural network based pattern matching and spike detection tools and services in the CARMEN neuroinformatics project,” Neural Networks, Special Issue on Neuroinformatics, 2008, vol. 21, Issue 8, pp. 1076-1084.

[22] De Roure, D. and Goble, C., “myExperiment: A web 2.0 virtual research environment for research using computation and services,” In: Workshop On Integrating Digital Library Content with Computational Tools and Services at JCDL, 2009.

[23] T. Finin, A. Joshi, L. Kagal, J. Niu, R. Sandhu, W. Winsborough, et al., “ROWLBAC: Representing role based access control in OWL,” SACMAT 08, 2008, pages 73-82.

[24] L. Zhao, S. Liu, J. Li, and H Xu, “A dynamic access control model based on trust,” 2nd Conference on Environmental Science and Information Application Technology, 2010, vol. 1, pages 548-551.

[25] A. Smirnov, A. Kashevnik, N. Shilov, and N. Teslya, “Context-based access control model for smart space,” 5th International Conference on Cyber Conflict (CyCon), 2013, pages 1-15.

[26] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham, “A semantic web based framework for social network access control,” Proc. of the 14th ACM symp. on Access control models and technologies, 2009, pp. 177-186.

[27] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham, “Semantic web-based social network access control,” Comp. & Security, March–May 2011, vol. 30, issues 2–3, pp. 108–115.

[28] British Atmospheric Data Centre (2013), “Homepage,” http://badc.nerc.ac.uk/, [retrieved: 11th April, 2014].

[29] Chou, W, “Inside SSL: the secure sockets layer protocol,“ IEEE Computer Society, 2002, ITPro, 47-52.

[30] C. Lu and S. Tseng, “Integrated design of AES (Advanced Encryption Standard) encrypter and decrypter,” Proceedings of The IEEE International Conference on Application-Specific Systems, Architectures and Processors, 2002, pp. 277–285.

[31] R. Buyya, R. Ranjan, and R. N. Calheiros, “InterCloud: utility-oriented federation of cloud computing environments for Scaling of application services,” 10th international conference on Algorithms and Architectures for Parallel Processing (ICA3PP), 2010, vol. part I, pages 13-31.

Intelligent BEE Method for Matrix-vector Multiplication on Parallel Computers

Seiji Fujino

Research Institute for Information
Technology, Kyushu University,
Fukuoka, Japan, 812-8581
E-mail: fujino@cc.kyushu-u.ac.jp

Yusuke Onoue

Kyushu DTS Ltd.,
Fukuoka, Japan 812-0011
E-mail: yusuke@zeal.cc.kyushu-u.ac.jp

George Abe

Graduate School of Information
Science and Electrical Engineering,
Kyushu University,
Fukuoka, Japan 812-8581
E-mail: g.abe@zeal.cc.kyushu-u.ac.jp

Abstract—**This paper compares the performance of sparse Matrix-vector multiplication paralleled by the conventional Block-Cyclic distribution and its improved variant on parallel computer with shared memory. The underlying idea is to exchange nonzero entries of matrix assigned to each thread with block unit. Numerical results demonstrate that the proposed distribution using exchange nonzero entries of matrix with block unit gives or improves parallelism.**

***Keywords*–Block exchanging; Matrix-vector multiplication; iBEE method; Parallel computers**

I. Introduction

We consider the problem of efficient Matrix-vector multiplication on parallel computers. As you know well, Matrix-vector multiplication appears often in solution of linear system of equations, and its efficient computation is crucial. In particular, Matrix-vector multiplication has a large part of computation of solving linear system of equations on parallel computers [10][11]. Many studies on fast computation of Matrix-vector multiplication have been proposed [3][6][7][12][13]. Fast computation depends greatly on evenly distributed nonzero entries of matrix onto each thread or process. In general, Block Cyclic (BC) distribution [8] is to be effective approach in order to evenly distribute nonzero entries of matrix. However, in the BC distribution, it is well known that parallel performance changes greatly as treated number of blocks changes. Therefore, in the BC distribution, it is not easy to decide optimum number of blocks.

In this paper, we propose an intelligent approach which distributes evenly nonzero entries of matrix to each thread by means of blocking exchange. We refer to intelligent Blocking Exchange for Evenly distributed nonzero entries of matrix (iBEE) method. We adopt double strategies for the iBEE method based on the conventional BC distribution. The first strategy is to decide the number of blocks so as to be evenly distributed for nonzero entries of matrix on each thread. The second strategy is to adopt a blocking exchange technique for refined and sufficiently even distribution. As a result, the iBEE method makes nonzero entries able to be significantly evenly distributed on each thread with the BC distribution.

The paper is organized as follows. In Section 2, we introduce a brief outline of the conventional Block and BC distributions. In Section 3, we describe our proposed iBEE method in detail. The iBEE method includes double strategies for the purpose of fast computation of the Matrix-vector multiplication on parallel computers. Moreover, in Section 4, we evaluate effectiveness of the iBEE method through numerical experiments. Finally, in Section 5, we will make concluding remarks.

II. The conventional distribution methods

We assume that nonzero entries are stored in Compressed Row Storage (CRS) format [2] and the pseudo program of computation of Matrix-vector multiplication is written in Fortran 90 [1]. Here, matrix A is sparse. In this case, concerning the conventional method for nonzero entries, the Block and BC distributions exist as simple distribution. Below, we give an outline of the Block and BC distributions. "$ncol$" means dimension of matrix, and "$rowptr$", "$colind$" and "val" mean arrays for starting pointer of each row, column index of each element and value of nonzero entries, respectively.

Pseudo program 1: Matrix-vector multiplication [2]

```
Do $i = 1, ncol$
   $temp = 0.0$
   Do $j = rowptr(i), rowptr(i+1) - 1$
      $temp = temp + val(j) * x(colind(j))$
   End Do
   $y(i) = temp$
End Do
```

A. Block distribution

In block distribution, we divide nonzero entries into blocks with the same number of threads. Moreover, we divide also nonzero entries such that the number of matrix row in each block is same each other.

In Fig.1 we show an example of two block distribution for matrix with dimension of 8 and with nonzero entries of 19. In this case, the difference of number of nonzero entries included in each block is three. Here, we set the thread number as "nth" and the number of blocks as "$nblk$". In block distribution, we get that $nblk = nth$.

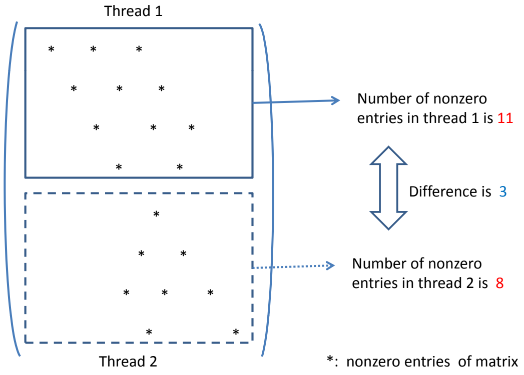

Thread 1
Number of nonzero entries in thread 1 is 11
Difference is 3
Number of nonzero entries in thread 2 is 8
Thread 2
*: nonzero entries of matrix

Fig.1 An example of block distribution in case of two threads.

We exhibit pseudo program for production of array of bst which stores the first row in each block.

Pseudo program 2: Production of array of bst [2]

```
$bst(1) = 1$
$tmp1 = ncol/nblk$
$tmp2 = \mathrm{mod}(ncol,\ nblk)$
Do $i = 1, tmp2$
   $bst(i+1) = bst(i) + tmp1$
End Do
Do $i = tmp2 + 1, nblk$
   $bst(i+1) = bst(i) + tmp1 + 1$
End Do
```

B. Block Cyclic distribution

We store block-id which is assigned to a thread to an array of asb (=assigned block). That is, we store block-id of the jth of block, which is assigned to the ith thread, to array of $asb(i, j)$. We produce an array of asb in the BC distribution as below.

Pseudo program 3: Production of array of asb in the BC distribution [2]

```
Do $i = 1, nth$
   Do $j = 1, nblk/nth$
      $asb(i,j) = i + nth * (j - 1)$
   End Do
End Do
```

Below, we present a parallel version of Matrix-vector multiplication with the OpenMP library [4][9] in the BC distribution. The array of bst stores row number on the starting row of each block. "$ncol$" means dimension of matrix, and "$rowptr$", "$colind$" and "val" mean arrays for starting pointer of each row, column index of each element and value of nonzero entries, respectively. "omp parallel do" means a directive for thread parallelism with the OpenMP library.

Pseudo program 4: Parallel version of Matrix-vector multiplication with OpenMP.

```
!$omp parallel do private($i, j, k, l, tmp, temp$)
Do $i = 1, nth$
   Do $j = 1, nblk/nth$
      $tmp = asb(i,j)$
      Do $k = bst(tmp), bst(tmp+1) - 1$
         $temp = 0.0$
         Do $l = rowptr(k), rowptr(k+1) - 1$
            $temp = temp + val(l) * x(colind(l))$
         End Do
         $y(k) = temp$
      End Do
   End Do
End Do
```

We present an example of the BC distribution in case of two threads in Fig.2. The number of nonzero entries in thread 1 is nine, and the number of nonzero entries in thread 2 is ten. In this case, the difference of number of nonzero entries included in each block is only one.

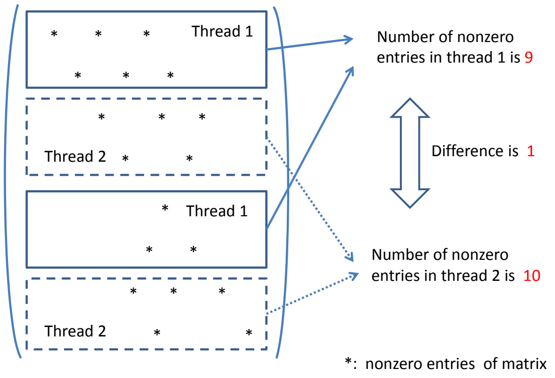

Thread 1
Thread 2
Thread 1
Thread 2
Number of nonzero entries in thread 1 is 9
Difference is 1
Number of nonzero entries in thread 2 is 10
*: nonzero entries of matrix

Fig.2 An example of the BC distribution in case of two threads.

III. Intelligent BEE method

In this section, we propose the iBEE method. The iBEE means intelligent blocking exchange technique for evenly distributed nonzero entries of matrix. The iBEE method is constructed based on the BC distribution, and adopts the following two intelligent strategies.

1) To determine the number of blocks automatically.
2) To apportion nonzero entries evenly with block exchanging technique.

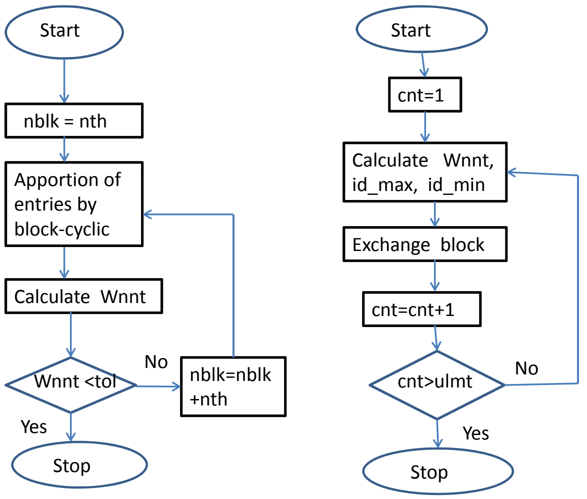
A. To determine automatically the number of blocks

In order to determine the number of blocks automatically, we introduce indicator $Wnnt$ (Width of nnt). $Wnnt$ is defined as follows:

$$\begin{aligned} \text{Wnnt} \;\; := \;\; & \max_{i \text{ in thread}}(nnt(i)) - \min(nnt(i)) \\ & (1 \le i \le nth). \end{aligned} \tag{1}$$

Here, "nth" means the thread number and "nnt" means the number of nonzero entries per thread.

Fig.3 (a) shows an algorithm to automatically compute the number of blocks per thread. In Fig.3, "$nblk$" means the number of blocks. At first, $nblk$ is initialized to nth. Next, we calculate indicator $Wnnt$, and check if $Wnnt < tolerance$ or not. If $Wnnt < tolerance$ then $nblk$ is determined to nth. On the other hand, if $Wnnt \ge tolerance$ then increase $nblk$ by nth. Until $Wnnt < tolerance$, $nblk$ is increased by nth.



(a)Determination of number of blocks (b)Exchanging blocks

Fig.3 Algorithm to automatically compute the number of blocks per thread.

B. To apportion evenly nonzero entries with block exchanging technique

Fig.3 (b) shows the algorithm of exchanging blocks. In Fig.3 (b), "u_lmt" means upper limit of the number of blocks exchanging. "id_max" means thread ID of the thread most apportioned nonzero entries and "id_min" means thread ID of the thread least apportioned nonzero entries. In Fig.3 (b), at first, cnt is initialized to one. Next, we calculate $Wnnt$ and id_max, id_min. Furthermore, the block apportioned to id_max and the block apportioned to id_min are exchanged. We increase cnt by one, and if $cnt > u\_lmt$ then the block exchange is finished.

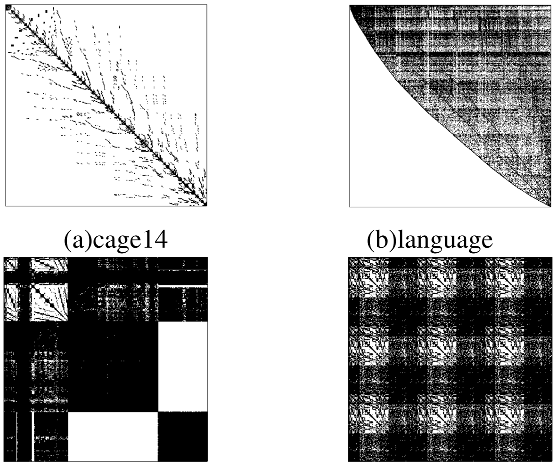
IV. Numerical Experiments

In this section we discuss numerical experiments of the BC distribution and the iBEE method. All computations were carried out in double precision floating-point arithmetic on FUJITSU PRIMEQUEST 580 (clock: 1.6GHz). FUJITSU optimized Intel Fortran Compiler90 and compile option "-Kfast, OMP" were used. We implemented all programs with the OpenMP library. The thread numbers are 1, 2, 4, 8, 16, 24, 32, 48 and 64. We set parameters of the iBEE method as $tolerance = 10000$ and u_lmt is the same as the thread number. Four test matrices are taken from Florida Sparse Matrix Collection [5]. The description of test matrices is shown in Table I. In this Table, "nnz" means number of nonzero entries, and "ave_nnz" means number of nonzero entries per single row. Moreover, "ave_nnz8" means average number of total nonzero entries per eight threads.

TABLE I. The description of test matrices.

matrix	dimension	nnz	ave_nnz	ave_nnz8	analytic field
cage14	1,505,785	27,130,349	18.02	3,391,294	DNA electrophoresis
language	399,130	1,216,334	3.04	152,041	language processing
poisson3Db	85,623	2,374,949	27.74	296,869	structural
sme3Dc	42,930	3,148,656	73.34	393,582	

Fig.4 shows the structure of four matrices. That is, Fig.4 plots nonzero entries of matrices. From Fig.4, we can see that a lower row decreases the number of nonzero entries of matrix language. Therefore, it is difficult to apportion sufficiently nonzero entries of matrix language evenly when we adopt the BC distribution. It is also difficult to apportion nonzero entries of matrix cage14 because the number of nonzero entries of matrix cage14 is very large.



(a)cage14 (b)language

(c)poisson3Db (d)sme3Dc

Fig.4 Pattern of nonzero entries of four matrices.

We present differences between minimum and maximum of nonzero entries when the thread numbers are 8 and 64 in

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-339-1

TABLE II. DIFFERENCE BETWEEN MINIMUM AND MAXIMUM OF NONZERO ENTRIES WHEN THE THREAD NUMBERS ARE 8 AND 64.

number of threads	matrix	ave. nnz per thread	(a)diff. of BC	(b)diff. of iBEE	ratio (=(b)/(a))
8	cage14	3,391,294	303,593 (8.95%)	7,028 (0.21%)	1/42.6
	language	152,041	35,626 (23.4%)	8,121 (5.34%)	1/4.38
	poisson3Db	296,869	19,088 (6.43%)	1,511 (0.51%)	1/12.6
	sme3Dc	393,582	89,719 (22.8%)	7,923 (2.01%)	1/11.3
64	cage14	423,911	173,813 (41.0%)	8,942 (2.11%)	1/19.4
	language	19,005	13,099 (68.9%)	9,362 (49.3%)	1/1.40
	poisson3Db	37,109	13,636 (36.7%)	9,654 (26.0%)	1/1.41
	sme3Dc	49,198	61,864 (125.7%)	6,638 (13.5%)	1/9.32

Table II. We see that the difference of the iBEE method is much smaller than that of the BC distribution at both 8 and 64 processors.

A. Computational cost of the iBEE method

We define D' as a difference between minimum and maximum of nonzero entries after exchanging blocks. We denote m as exchanging block-id included in minimum nonzero entries and M as that included in maximum nonzero entries, respectively. Moreover, we denote $bnz(m)$ and $bnz(M)$ as number of nonzero entries with block-id of m and that of nonzero entries with block-id of M, respectively. Then, the difference D' is written as

$$D' = |W_{\rm nnt} - 2(bnz(M) - bnz(m))|. \tag{2}$$

That is, first we calculate the above difference of D' to all combinations of block exchanging, secondly we may exchange blocks so as to be minimum nonzero entires for difference D'. For example, when number of blocks included in each thread is 100, number of combination of block exchanging is estimated as only $100 \times 100 = 10,000$. Then, we may calculate the above difference D' at 10000 times. Therefore, we can do it quickly. As a result, we can estimate that the cost of iBEE method is not expensive at all, because there is no reordering of nonzero entries of matrix.

We ran two experiments. The first experiment is performance estimation of parallel Matrix-vector multiplication with the BC distribution and the iBEE method.

B. Performance estimation of parallel Matrix-vector product with block-cyclic distribution and the iBEE method

In this section, we show numerical results of parallel Matrix-vector product with the BC distribution and the iBEE method. We tested Matrix-vector multiplication at 1000 times. Table III shows the performance of the iBEE method. In Table III, "nth" means the thread number and "$nblk$" means the number of blocks. Bold figures means minimum total time among the BC distribution and the iBEE method.

From Table III, we can see that $Wnnt$ of the iBEE method is much smaller than that of the BC distribution, and time of the iBEE method is shorter than that of the BC distribution for all thread number. In particular, the iBEE method works well when the thread number becomes larger than 32 threads. Moreover, it is concluded that the iBEE method is very effective for matrix sme3Dc.

TABLE III. COMPARISON OF PERFORMANCE OF PARALLELED MATRIX-VECTOR MULTIPLICATION WITH THE BC DISTRIBUTION AND THE iBEE METHOD.

(a)matrix: cage14

			$Wnnt$		time[s]		speed-
method	nth	$nblk$		ratio(%)	$A\boldsymbol{v}$-t	total-t	up
BC iBEE	1	1	0	-	184.598	184.598	1.0
BC	2	10	1,233,727	100.0	101.987	101.987	1.81
iBEE			6,633	0.54	101.423	**101.423**	1.82
BC	4	28	745,785	100.0	71.941	71.941	2.56
iBEE			7,435	1.00	71.606	**71.606**	2.57
BC	8	104	303,593	100.0	38.499	38.499	4.79
iBEE			7,028	2.31	37.522	**37.522**	4.92
BC	16	96	438,366	100.0	26.635	26.635	6.93
iBEE			9,191	2.10	25.127	**25.127**	7.34
BC	24	120	321,120	100.0	18.538	18.538	9.95
iBEE			9,748	3.04	17.974	**17.974**	10.27
BC	32	224	251,881	100.0	11.688	11.688	15.79
iBEE			8,272	3.28	10.993	**10.993**	16.79
BC	48	288	193,305	100.0	3.863	3.863	47.78
iBEE			6,300	3.26	3.212	**3.212**	57.47
BC	64	256	173,813	100.0	2.340	2.340	78.88
iBEE			8,942	5.14	2.138	**2.138**	86.34

(b)matrix: language

			$Wnnt$		time[s]		speed-
method	nth	$nblk$		ratio(%)	$A\boldsymbol{v}$-t	total-t	up
BC iBEE	1	1	0	-	20.382	20.382	1.0
BC	2	10	112,544	100.0	8.964	8.964	2.27
iBEE			2,078	1.85	8.793	**8.793**	2.31
BC	4	28	54,286	100.0	4.034	4.034	5.05
iBEE			4,147	7.64	4.008	**4.008**	5.08
BC	8	280	35,626	100.0	2.007	2.007	10.15
iBEE			8,121	22.80	1.964	**1.965**	10.37
BC	16	368	28,770	100.0	1.046	1.046	19.48
iBEE			9,623	33.45	1.012	**1.013**	20.12
BC	24	600	27,922	100.0	0.744	0.744	27.39
iBEE			9,453	33.86	0.702	**0.703**	28.99
BC	32	800	21,982	100.0	0.602	0.602	33.85
iBEE			9,802	44.59	0.554	**0.556**	36.65
BC	48	1008	15,896	100.0	0.449	0.449	45.39
iBEE			9,916	62.38	0.421	**0.422**	48.29
BC	64	1152	13,099	100.0	0.391	0.391	52.12
iBEE			9,362	71.47	0.354	**0.355**	57.41

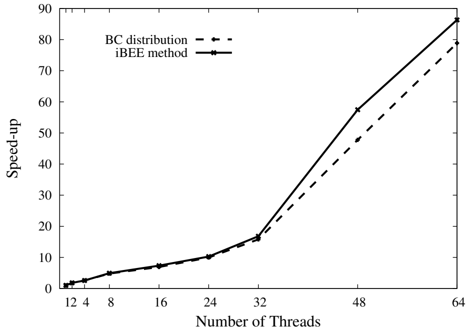
(c)matrix: poisson3Db

			$Wnnt$		time[s]		speed-
method	nth	$nblk$		ratio(%)	$\boldsymbol{Av}$-t	total-t	up
BC	1	1	0	-	14.921	14.921	1.0
iBEE							
BC	2	14	10,581	100.0	6.853	6.853	2.17
iBEE			665	6.28	6.707	**6.707**	2.22
BC	4	64	21,988	100.0	3.036	3.036	4.91
iBEE			1,607	7.31	3.015	**3.015**	4.94
BC	8	128	19,088	100.0	1.527	1.527	9.77
iBEE			1,511	7.92	1.517	**1.517**	9.83
BC	16	144	33,619	100.0	0.833	0.833	17.91
iBEE			7,705	22.92	0.789	**0.787**	18.95
BC	24	216	21,758	100.0	0.573	0.573	26.04
iBEE			5,570	25.56	0.545	**0.542**	27.53
BC	32	256	11,863	100.0	0.434	0.435	34.30
iBEE			9,004	75.90	0.424	**0.424**	35.19
BC	48	384	8,781	100.0	0.318	0.318	46.92
iBEE			6,136	69.88	0.313	**0.313**	47.67
BC	64	384	13,636	100.0	0.296	0.296	50.40
iBEE			9,654	70.80	0.273	**0.273**	54.65

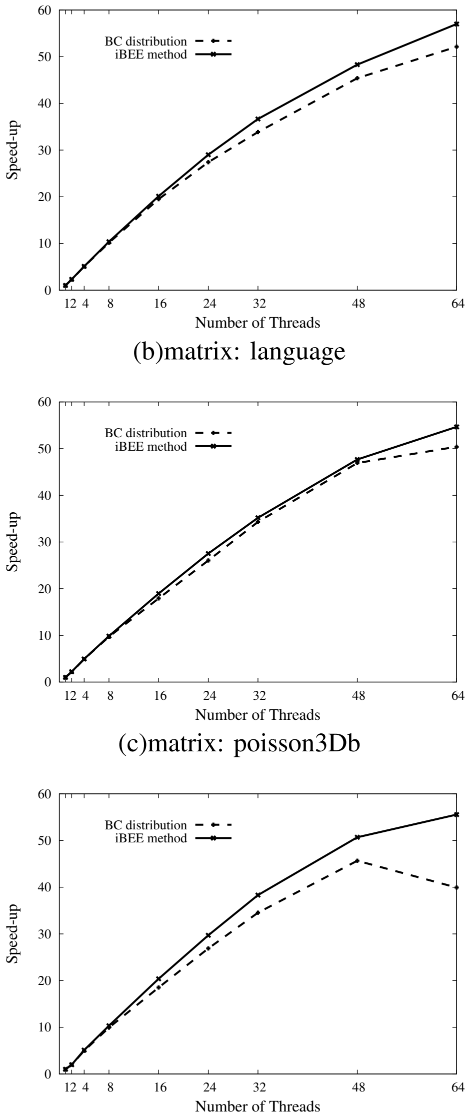
(d)matrix: sme3Dc

			$Wnnt$		time[s]		speed-
method	nth	$nblk$		ratio(%)	$\boldsymbol{Av}$-t	total-t	up
BC	1	1	0	-	13.335	13.335	1.0
iBEE							
BC	2	8	76,358	100.0	6.719	6.719	1.98
iBEE			6,330	8.29	6.702	**6.702**	1.99
BC	4	20	80,777	100.0	2.680	2.680	4.97
iBEE			4,839	5.99	2.606	**2.606**	5.11
BC	8	40	89,719	100.0	1.344	1.344	9.92
iBEE			7,923	8.83	1.291	**1.291**	10.32
BC	16	112	41,814	100.0	0.720	0.720	18.52
iBEE			3,772	9.02	0.654	**0.654**	20.39
BC	24	168	28,798	100.0	0.496	0.496	26.88
iBEE			3,080	10.70	0.449	**0.449**	29.69
BC	32	224	24,769	100.0	0.386	0.386	34.54
iBEE			1,873	7.56	0.348	**0.348**	38.31
BC	48	336	16,990	100.0	0.292	0.292	45.66
iBEE			1,505	8.86	0.263	**0.263**	50.70
BC	64	384	61,864	100.0	0.334	0.334	39.92
iBEE			6,638	10.73	0.240	**0.240**	55.56

Fig.5 (a)-(d) plots the speed-up of parallel Matrix-vector multiplication with the BC distribution and the iBEE method. In Fig.5 (a)-(d), dashed line plots speed-up of the BC distribution and solid line plots that of the iBEE method. From Fig.5 (a)-(d), speed-up of the iBEE method is larger than that of the BC distribution.



(a)matrix: cage14



(b)matrix: language

(c)matrix: poisson3Db

(d)matrix: sme3Dc

Fig.5 Comparison of speed-up of paralleled Matrix-vector multiplication with the BC distribution and the iBEE method.

In Fig.6, we show the tendency of ratio (%) of $Wnnt$ when the thread number changes. We see that ratios of $Wnnt$ are very low for four matrices compared with the BC distribution.

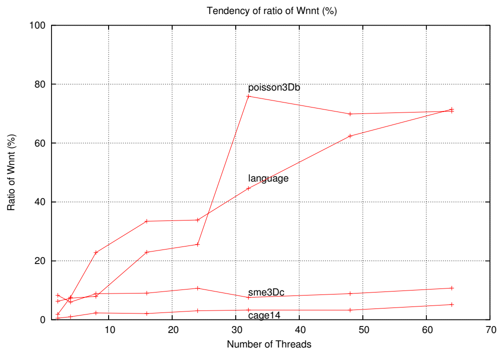


Fig.6 Tendency of ratio (%) of $Wnnt$ when the thread number changes.

V. Concluding Remarks

We proposed an intelligent Blocking exchange technique of Evenly distributed for nonzero Entries of matrix. Moreover, we evaluated the performance of parallel Matrix-vector product using the iBEE method. As a result, it turned out that the iBEE method can distribute nonzero entries of matrix more evenly than the conventional BC distribution. Moreover, parallel performance of Matrix-vector multiplication with the iBEE method is faster than that with the BC distribution.

References

[1] C. Arai, Introduction of Fortran90, Morikita Corp., 1998.

[2] B. Barrett, et al., Templates, SIAM, Philadelphia, 2000.

[3] R. H. Bisseling and W. Meesen, "Communication balancing in parallel sparse matrix-vector multiplication", Electronic Trans. on Numerical Analysis, vol.21, pp.47–65, 2005.

[4] R. Chandra, et al., Parallel programming in OpenMP, Morgan Kaufmann Publishers, 2001.

[5] T. Davis, Univ. of Florida sparse matrix collection, http://www.cise.ufl.edu/research/sparse/matrices/index.html [accessed: 2014-04-08].

[6] E. J. Im, K. Yelick and R. Vuduc, "Sparsity: Optimization framework for sparse matrix kernels", Int. J. of High Performance Computing Applications, vol.18, no.1, pp.135–158, 2004.

[7] S. Lee and R. Eigenmann, "Adaptive Runtime Tuning of Parallel Sparse Matrix-Vector Multiplication on Distributed Memory Systems", The Proc. of the 22nd Annual Int. Conference on Supercomputing, June 2008, pp.195–204.

[8] J. J. Lo, et al., "Tuning Compiler Optimizations for Simultaneous Multithreading", Int. J. of Parallel Programing, vol.27, no.6, pp.114–124, 1999.

[9] T. G. Mattson, B. A. Sanders and B. L. Massinggill, Pattern for Parallel programing, Addison Wesley, 2005.

[10] Y. Saad, Iterative methods for sparse linear systems 2nd edition, SIAM, Philadelphia, 2003.

[11] H. A. van der Vorst, Iterative Krylov methods for large linear systems, Cambridge University Press, Cambridge, 2003.

[12] S. Williams, et al., "Optimization of sparse matrix-vector multiplication on emerging multicolor platforms", Parallel Computing, vol.35, pp.178–194, March 2009.

[13] A. N. Yzelman and R. H. Bisseling, "Cache-oblivious sparse matrix-vector multiplication by using sparse matrix partitioning methods", SIAM, J. on Scientific Computing, vol.31, no.4, pp.3128–3154, 2009.

ISBN: 978-1-61208-339-1

A Network Architecture for Distributed Event-Based Systems in an Ubiquitous Sensing Scenario

Cristina Muñoz, Pierre Leone
Department of Computer Science
University of Geneva
Carouge, Switzerland
Email: {Cristina.Munoz, Pierre.Leone}@unige.ch

***Abstract*—Ubiquitous sensing devices frequently disseminate their data between them. The use of a distributed event-based system that decouples publishers of subscribers arises as an ideal candidate to implement the dissemination process. In this paper, we present a network architecture which merges the network and overlay layers of typical structured event-based systems. Directional Random Walks (DRWs) are used for the construction of this merged layer. Our first results show that DRWs are suitable to balance the load using a few nodes in the network to construct the dissemination path. As future work, we propose to study the properties of this new layer and to work on the design of Bloom filters to manage broker nodes.**

***Keywords*-Distributed Event-Based systems; Directional Random Walks; Bloom filters; Wireless Sensor Networks.**

I. Introduction

Ubiquitous or pervasive computing [1] uses many sources and destinations to gather and process data related to physical processes with the aim of making possible human-computer interaction. In the process of dissemination, some devices generate the data, while others are waiting for the sensing data. In this context, the use of a distributed event-based system [2] arises as an ideal candidate to implement the model of communication on the reception or transmission of events.

The main characteristic of an event-based system is that publishers and subscribers are decoupled. This means that they do not have any information about each other. The element in charge of matching notifications with subscriptions is called the event notification service. In distributed networks, the event notification service is implemented using a network of brokers nodes (see Figure 1). It is considered that a broker is any node in the network that has information about any single or set of subscriptions. The complexity of designing this type of systems usually lies on the way to elect the nodes which will act as brokers because of the decentralized nature of a distributed network.

In our research, we assume that a node can be a publisher, a subscriber, a broker or a combination of these three possibilities. We also assume that all the nodes in the network are able to participate in it without the requirement to adopt the specific role of publisher or subscriber. Nodes that are actively participating in the network but do not take any specific role will be considered as part of the overlay layer. Those nodes of the overlay layer that are able to redirect messages will be considered as brokers.

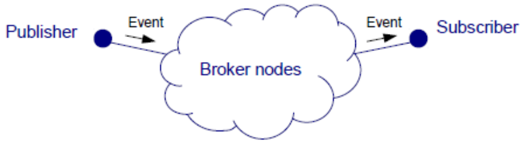


Figure 1: Distributed notification service using a network of brokers.

Event-based systems are classified as topic-based or content-based [2]. Topic-based systems take into account the subject of messages in order to match publications with subscriptions. Content-based systems use filters to specify the value of subscriptions' attributes to redirect notifications. A filter is a boolean function built taking into account the set of subscriptions. In our proposal, we plan to deal with a content-based system that uses Bloom filters at broker nodes in order to save memory resources and speed up routing decisions.

Sensor networks frequently use tiny devices with limited battery capabilities that make unsuitable the use of a Global Positioning System (GPS) to disseminate information according to the coordinates of nodes. In addition to this, the use of virtual coordinates to substitute real coordinates requires the use of sinks or landmarks to structure the network. For these reasons, the use of coordinates in an unstructured sensing scenario is not recommended. We assume that we work in an unstructured scenario in which no routing protocol provides communication between the nodes of the network.

The constraints of the network's infrastructure lead us to the design of a network architecture for distributed event-based systems that must use as less resources as possible (i.e., battery, memory, etc.). In this paper, we present a solution that avoids implying all the nodes of the network in the dissemination process by using a distributed notification service defined by Directional Random Walks (DRWs).

The rest of this paper is organized as follows: Section II analyzes the state of the art. Section III points out the approach to solve the problem specified in this section. Section IV presents the research efforts already done for the approach specified in Section III. Finally, Section V summarizes our proposal and suggests further work.

II. State of the art

A. Distributed and Structured Event-based Systems

Distributed and structured event-based systems use three layers on the top of a bottom layer (see Figure 2), which provides data link functionalities, to facilitate topology control:

1) The network layer is in charge of providing data forwarding between the different nodes involved in the network. A network protocol, such as the Multicast Ad-hoc On-demand Distance Vector (MAODV) [3] is needed to provide point-to-point communication.
2) The medium layer is called the overlay layer. It is a virtual layer that builds the event notification service by providing a network of brokers that redirect notifications to the corresponding subscribers.
3) Finally, on the top layer the event-based protocol is implemented.

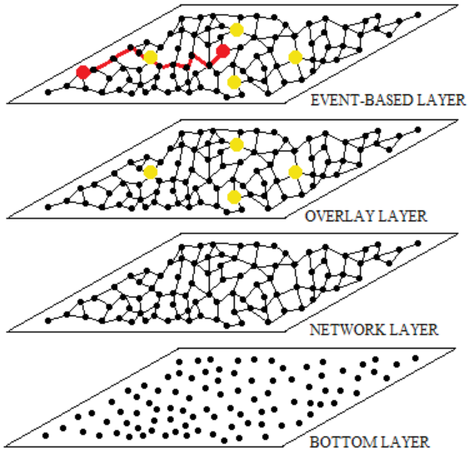


Figure 2: Decomposition in layers of the typical design of a distributed and structured event-based system.

One strategy to construct the overlay layer is to use a tree. In TinyMQ [4], which is designed specifically for wireless sensor networks, a multi-tree overlay layer is maintained.

Another strategy is to clusterize the network and use cluster heads to manage messages as in Mires [5], which is a middleware for sensor networks. The Gradient Landmark-Based Distributed Routing (GLIDER) [6] organizes the network using some defined landmarks to compute the Delaunay graph for network partition. Then, the Landmark-Based Information Brokerage scheme (LBIB) [7] uses an overlay layer based in GLIDER to match publishers with subscribers.

A typical solution is to build the overlay layer using Distributed Hash Tables (DHTs). In these systems, a key is mapped to a particular node with storage location properties. In some DHT architectures, rendez-vous nodes depend on the node ID as in Pastry [8]. In others, as the Content Addressable Network (CAN) [9], a region of the space is used to map a key. Some efforts have been made to apply this solution to sensor networks [10]. When coordinates are available, sensor networks use Geographic Hash Tables (GHT) instead of a typical DHT. Currently, technology companies as Ericsson Research, are making an effort to develop applications that use GHTs in wireless sensor networks [11].

B. *Distributed and Unstructured Event-based Systems*

The main characteristics of distributed and unstructured event-based systems is that they do not maintain an overlay layer. This fact makes easier to deal with network's changes. The distributed notification service may be built using flooding, gossiping or random walks.

Most of the algorithms proposed deal with the unstructured nature of wireless communications using flooding to build a tree. A typical solution is to use the On-Demand Multicast Routing Protocol (ODMRP) [12], which is based in the forwarding group concept. Groups are constructed and maintained periodically when a multicast source has data to send. This task is done by broadcasting the entire network with membership information. An extension for ODMRP has been proposed [13] to adapt a content-based system by adding subscriptions to Bloom filters. Trees also may be configured to self-repair themselves in base to brokers' dynamicity [14]. These solutions are reliable but increase the traffic of the network because they use flooding at some point.

Flooding may also be used to continuously exchange subscription information clusterizing the network [15]. Then, notifications are sent to the appropriate cluster, improving the efficiency of the network. Other mechanisms can be used as the combination of a DHT and random walks [16]. Cluster heads manage the DHTs while random walks help to connect the different cluster heads of the network. The cluster concept in the network of brokers can be improved in a dynamic scenario by enriching the topology management with predictions based on location [17].

1) Probabilistic approaches: Probabilistic approaches are suitable to deal with dynamicity but they do not offer reliability. Some solutions propose that all the nodes in the network implement a broker that forwards messages to neighbors depending on the estimation of potential subscribers [18]. Other solutions [19], propose to flood subscriptions in a small area and then use random walks to reach that areas. In Quasar [20], subscriptions of a certain area are able to attract or reject notifications, that are propagated with a random walk, using an attenuated Bloom filter [21]. A probabilistic solution that uses a random walk specifically designed to go deep into the network is CoQUOS (Continous Queries on Unstructured OverlayS) [22]. Continuous queries are launched to the network using random walks. Peers compute the overlap between their neighbor's lists and use this information to forward the random walk to avoid remaining in a cluster. Then, some peers register the query with a probability that depends on the number of hops.

III. Network Architecture

Due to the unstructured nature of our network, we propose the development of a dissemination algorithm that merges the network and the overlay layers of a typical distributed and structured event-based system (see Figure 2). This means that no network protocol is needed. The main advantage of not using a network protocol is that there is no necessity of maintaining a network topology. This implies that most nodes of the network, which do not actively participate in the process of dissemination, do not have to keep any information about topology. The main consequence is that nodes not involved in the system are able to save energy and computing resources.

Our design (see Figure 3) uses two layers on the top of a bottom layer that provides data link functionalities:

1) The overlay layer is in charge of providing the distributed network of brokers and, at the same time, provides point-to-point communication between publishers and subscribers. The main objective of this strategy is to avoid the use of global information of the network which is costly to get and maintain.
2) As in Figure 2, the event-based protocol is implemented at the top layer.

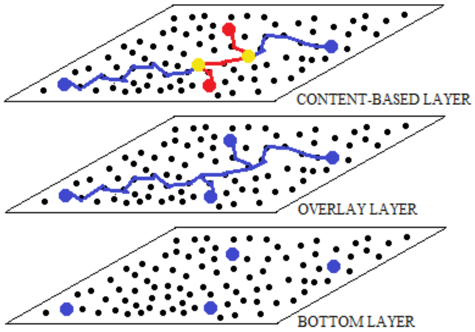


Figure 3: Decomposition of the architecture of our design in layers.

As Section I mentions, we assume that a node can be a publisher, a subscriber, a broker or a combination of these three possibilities. Moreover, our design takes advantage of some nodes in the network which want to collaborate. Nodes that participate in the system are considered part of the overlay layer (blue path of Figure 3). The overlay layer is formed by the intersection of different publishers and subscribers (blue nodes). Publishers and subscribers implement a DRW until intersecting other DRW. Broker nodes (yellow nodes) are the meeting point between two DRWs.

A DRW is a probabilistic technique able to go forward into the network following a loop-free path. The principle assumed in this strategy is that two lines in a plane cross (see Figure 4). It is unclear how to construct a straight path of relaying nodes in ubiquitous unstructured networks without requiring global information and without making use of geo-coordinates. In this research, two different methods have been proposed in order to build DRWs [23][24].

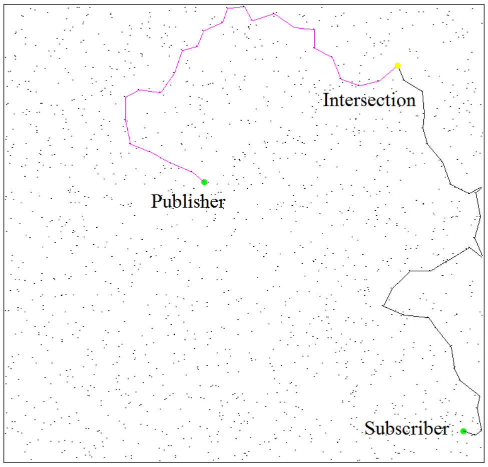


Figure 4: Directional Random Walks intersecting using a Java simulator.

The matching of publishers and subscribers will be done using a special architecture of Bloom filters [21] implemented at broker nodes. Bloom filters are probabilistic data structures that efficiently manage membership of a certain number of elements. The content related to membership is hashed using different hashing algorithms. Then, the positions of the Bloom structure corresponding to the hashes are set to one. The maximum number of elements to be inserted to the filter is fixed in order to maintain a certain probability of false positives. When searching for elements of a certain membership, the correspondent positions of the data structure are checked. The main advantage of Bloom filters is that they do not require much memory space and processing resources; so its use is very convenient in a sensing scenario in which devices have limited capabilities.

It is remarkable to mention that in our event-based system no advertisement table is required because filters just manage information about subscriptions.

IV. Research efforts

In this section, we present the efforts already made in order to build the overlay layer proposed.

Based on [23], a first method to build DRWs is proposed. It is based in the addition of different nodes to the DRW by pre-computing different weights at each node that take into account the two hops path. A weight is defined as follows:

$$n_{xz}^{y} = |\, N(x) \cap N(z)\, | \qquad (1)$$

where y is the last node added to the DRW; x is the penultimate node added to the DRW; z can be any node of the set $N(y)$ and $N(a)$ is the set of neighbor nodes of node a. Furthermore, in this method, a penalty is added to the weight when a node is added to the DRW.

Some properties about our heuristics were found using extensive simulations. The first property claims that DRWs decrease the time to intersection compared to pure random walks. The second property states that cooperation also decreases the time to intersection. Cooperation refers to synchronicity between publishers and subscribers. Finally, it is shown that DRWs are good at balancing the load of the network.

Based on [24], a second method to build DRWs is proposed. The main difference with the first design presented for DRWs is that nodes of the first and second neighborhoods of nodes added to the DRW are marked. In addition to this, the cost is not pre-computed, but it is computed when selecting a node as follows:

$$c(v) = \alpha|N(v) \cap N(DRW)| + \beta|N(v) \cap N^2(DRW)| \quad (2)$$

where α and β are parameters used as weights; v can be any node of the set related to the neighborhood of the last node added to the DRW; DRW is the set of nodes that are part of the DRW; $N(a)$ is the set of neighbor nodes of node a and $N^2(DRW)$ is the set of neighbor nodes of $N(DRW)$.

In the first part of this research, the properties associated with a DRW were assessed. Implementations of DRWs of one branch or two branches were studied. The main results show that the use of one branch is as efficient as the use of two branches. Moreover, it is shown that the use of second neighborhoods to forward the DRW does not improve the Euclidean distance traversed in the network. It is also shown that shorter paths are obtained when using higher densities of nodes in the network. In the second part of this research, an information brokerage system was evaluated using a double ruling method. As in the first paper, it is shown that the algorithm is good at balancing the load using a few nodes of the network. In fact, we can state that the method proposed is as good as a traditional Rumor Routing algorithm [25] with an infinite memory.

V. Conclusion and Future work

In this paper, a novel network architecture for distributed event-based systems that use sensing devices has been proposed. Our first results, validated through extensive simulations, show that DRWs are suitable for the construction of an overlay layer that provides point-to-point communication and a distributed notification service. This is due, mainly to the good properties of DRWs to balance the load using a few nodes of the network for the establishment of paths.

Currently, we are working on the first simulations of the overlay layer to study the impact of having different number of publishers and subscribers. Figure 5 shows a simulation of the overlay layer in which yellow squares represent the distributed network of brokers while publishers and subscribers are represented using green circles.

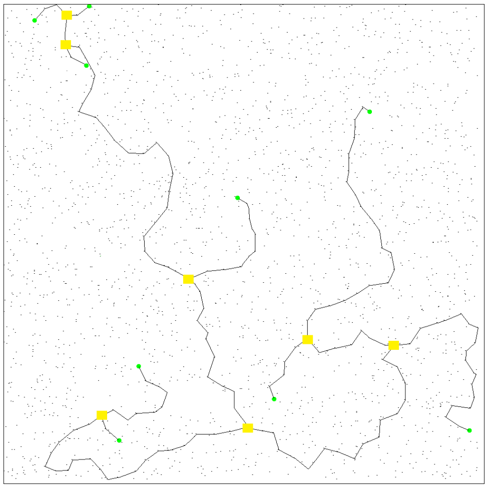


Figure 5: Overlay layer using Directional Random Walks in a Java simulator.

At the same time, we are working on the design of Bloom filters for broker nodes that will be specifically fitted for sensing constrained devices.

Finally, the deployment of a sensing testbed that will use wireless sensor nodes, will evaluate the suitability of the proposed solution under real conditions.

ACKNOWLEDGMENT

This work has been developed as part of the POPWiN project (Parallel Object Remote Programming for Heterogeneous Wireless Networks over IPv6) that is financially supported by the Hasler Foundation in its SmartWorld - Information and Communication Technology for a Better World 2020 program.

REFERENCES

[1] D. J. Cook and S. K. Das, “Review: Pervasive computing at scale: Transforming the state of the art,” Pervasive Mob. Comput., vol. 8, no. 1, Feb. 2012, pp. 22–35.

[2] G. Mühl, L. Fiege, and P. Pietzuch, Distributed Event-Based Systems. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[3] S. Roy, V. Addada, S. Setia, and S. Jajodia, “Securing maodv: attacks and countermeasures,” in Sensor and Ad Hoc Communications and Networks, 2005. IEEE SECON 2005. 2005 Second Annual IEEE Communications Society Conference on, Sept 2005, pp. 521–532.

[4] K. Shi, Z. Deng, and X. Qin, “Tinymq: A content-based publish/subscribe middleware for wireless sensor networks,” in SENSORCOMM 2011, The Fifth International Conference on Sensor Technologies and Applications, 2011, pp. 12–17.

[5] E. Souto et al., “Mires: a publish/subscribe middleware for sensor networks,” Personal and Ubiquitous Computing, vol. 10, no. 1, 2006, pp. 37–44.

[6] Q. Fang, J. Gao, L. J. Guibas, V. Silva, and L. Zhang, “Glider: Gradient landmark-based distributed routing for sensor networks,” in in Proc. of the 24th Conference of the IEEE Communication Society (INFOCOM, 2005, pp. 339–350.

[7] Q. Fang, J. Gao, and L. J. Guibas, “Landmark-based information storage and retrieval in sensor networks,” in In The 25th Conference of the IEEE Communication Society (INFOCOM06, 2006, pp. 1–12.

[8] A. I. T. Rowstron and P. Druschel, “Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems,” in Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg, ser. Middleware ’01. London, UK, UK: Springer-Verlag, 2001, pp. 329–350.

[9] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, “A scalable content-addressable network,” in Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, ser. SIGCOMM ’01. New York, NY, USA: ACM, 2001, pp. 161–172.

[10] G. Fersi, W. Louati, and M. Ben Jemaa, “Distributed hash table-based routing and data management in wireless sensor networks: a survey,” Wirel. Netw., vol. 19, no. 2, Feb. 2013, pp. 219–236.

[11] H. Mahkonen, T. Jokikyyny, J. Jiménez, and S. Kukliński, “M3: Machine-to-machine management framework,” in Proc. 3rd Intl. Conference on Sensor Networks, SENSORNETS, 2014, pp. 139–144.

[12] S. J. Lee, W. Su, and M. Gerla, “On-demand multicast routing protocol in multihop wireless mobile networks,” Mob. Netw. Appl., vol. 7, no. 6, Dec. 2002, pp. 441–453.

[13] E. Yoneki and J. Bacon, “An adaptive approach to content-based subscription in mobile ad hoc networks,” in Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, ser. PERCOMW ’04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 92–97.

[14] L. Mottola, G. Cugola, and G. P. Picco, “A self-repairing tree topology enabling content-based routing in mobile ad hoc networks,” IEEE Transactions on Mobile Computing, vol. 7, no. 8, Aug. 2008, pp. 946–960.

[15] S. Voulgaris, E. Rivire, A.-M. Kermarrec, and M. V. Steen, “Sub-2-sub: Self-organizing content-based publish subscribe for dynamic large scale collaborative networks,” in In IPTPS06: the fifth International Workshop on Peer-to-Peer Systems, 2006.

[16] R. Tian et al., “Hybrid overlay structure based on random walks,” in Proceedings of the 4th international conference on Peer-to-Peer Systems, ser. IPTPS’05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 152–162.

[17] F. Abdennadher and M. Ben Jemaa, “Accurate prediction of mobility into publish/subscribe,” in Proceedings of the 11th ACM International Symposium on Mobility Management and Wireless Access, ser. MobiWac ’13. New York, NY, USA: ACM, 2013, pp. 101–106.

[18] J. Haillot and F. Guidec, “Content-based communication in disconnected mobile ad hoc networks,” in Proceedings of the 8th international conference on New technologies in distributed systems, ser. NOTERE ’08. New York, NY, USA: ACM, 2008, pp. 21:1–21:12.

[19] P. Costa and G. Picco, “Semi-probabilistic content-based publish-subscribe,” in Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE International Conference on, 2005, pp. 575–585.

[20] B. Wong and S. Guha, “Quasar: a probabilistic publish-subscribe system for social networks,” in Proceedings of the 7th international conference on Peer-to-peer systems, ser. IPTPS’08. Berkeley, CA, USA: USENIX Association, 2008, pp. 2–2.

[21] S. Tarkoma, C. Rothenberg, and E. Lagerspetz, “Theory and practice of bloom filters for distributed systems,” Communications Surveys Tutorials, IEEE, vol. 14, no. 1, First 2012, pp. 131–155.

[22] L. Ramaswamy and J. Chen, “The coquos approach to continuous queries in unstructured overlays,” IEEE Trans. on Knowl. and Data Eng., vol. 23, no. 3, Mar. 2011, pp. 463–478.

[23] P. Leone and C. Muñoz, “Content based routing with directional random walk for failure tolerance and detection in cooperative large scale wireless networks,” in Proc. 2nd Intl. Workshop on Architecting Safety in Collaborative Mobile Systems, SAFECOMP, 2013, pp. 313–324.

[24] C. Muñoz and P. Leone, “Design of an unstructured and free geo-coordinates information brokerage system for sensor networks using directional random walks,” in Proc. 3rd Intl. Conference on Sensor Networks, SENSORNETS, 2014, pp. 205–212.

[25] D. Braginsky and D. Estrin, “Rumor routing algorthim for sensor networks,” in Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, ser. WSNA ’02. New York, NY, USA: ACM, 2002, pp. 22–31.

An Improvement on Acceleration of Distributed SMT Solving

Leyuan Liu, Weiqiang Kong, Takahiro Ando, Hirokazu Yatsu, and Akira Fukuda

Graduate School of Information Science and Electrical Engineering
Kyushu University, Japan
Email: leyuan@f.ati.kyushu-u.ac.jp, weiqiang@qito.kyushu-u.ac.jp,
{ando.takahiro, hirokazu.yatsu, fukuda}@ait.kyushu-u.ac.jp

Abstract—**Satisfiability Modulo Theories-based Bounded Model Checking consists of two primary tasks: (1) encoding a bounded model checking problem into a propositional formula that represents the problem, and (2) using a SMT solver to solve the formula. Solving the formula (namely, SMT solving) involves computation-intensive processes and is thus time-consuming. The target of this paper is to improve the distributed SMT solving techniques we previously proposed in [1] for further enhancing the effectiveness of SMT-based BMC. To this end, we improve the file dispatching scheme and reform the communication protocols in our previous work. In this paper, we describe the amelioration details and give a series of experiments to show the effectiveness of our improvement. Experimental results show that the improved implementation outperform the previous one. In addition to solving 8 groups of benchmarks by increasing the number of clients, we also make a preliminary experiment on increasing Central Processing Unit cores to investigate the influence.**

Keywords*–*Satisfiability Modulo Theories; Distributed Solving; Acceleration; MPI; OpenMP.

I. Introduction

Bounded Model Checking (BMC) is a restricted form of model checking [2] that analyzes if a desired property hold in bounded execution/behaviors of a system. In a nutshell, BMC can be explicit-state based BMC such as the methods described in [3] and symbolic-based BMC such as Binary Decision Diagram (BDD)-based [4], Boolean Satisfiability (SAT)-based [5] or SMT-based [6] BMC. It has been reported in [7] that symbolic-based methods perform better than explicit-based methods for verifying general Linear Temporal Logic (LTL) [2] properties. Among the symbolic-based BMC methods, the Satisfiability Modulo Theories (SMT)-based method is more expressible (thanks to its rich background theories) and is able to generate more compact formulas, and therefore, is more and more adopted by researchers and engineers.

SMT-based BMC consists of two primary tasks: (1) encoding a bounded model checking problem into a propositional formula that represents the problem, and (2) using a SMT solver to solve the formula, that is, finding a set of variable assignments that makes the formula true. Solving the formula (namely, SMT solving) involves computation-intensive processes and is thus time-consuming. Furthermore, as the model-checking bound increases, the encoded formulas become larger in size and harder to solve. The computational complexity of most SMT problems is very high [8], [9]. For all that, it is difficult to accelerate the SMT solving procedure for the engineers engaged in model checking. We have conducted [1] an implementation of using distributed computation and utilizing the power of multi-cores Central Processing Unit (CPU), multi-CPUs, and/or even cloud computing, to accelerate SMT solving. Although a series of experiments has shown the effectiveness of our implementation on increasing the solving efficiency, there still exist shortcomings which prevent the distributed solving to take advantage of CPU cores as much as possible. For example, the communication protocols could be reformed in order to reduce the unnecessary usage of network communication. In this paper, we describe our work on improving the distributed SMT solving by changing the file dispatching schemes that consider work load balance, and by reforming the communication protocols. We change file dispatching from coarse-grained to fine-grained, which can help in increasing the usage of CPU cores. Some unnecessary steps in the communication protocols are removed or merged. We have discussed the effectiveness of our improvement theoretically. We repeat the experiments conducted in our previous work [1] to make comparisons between these two schemes. We also conduct an experiment by increasing the CPU cores used in parallel SMT solving to investigate the influence in a microscopic view. The experimental results demonstrate the feasibility and efficiency of our improved implementation. However, we have also found, for a given target benchmark, that increasing CPU cores involved in computing will not always increase the solving speed.

The rest of this paper is structured as follows. Section II provides necessary preliminary knowledge and a brief introduction of the tools and techniques that are used in our work. Section III describes our previous work about distributed SMT solving. Section IV shows our methods used to improve the distributed SMT solving. Section V presents the experiments to evaluate the improvement and discusses the results. Finally, Section VI mentions possible extension (application scenarios) of our work and concludes the paper.

II. Preliminary Knowledge

A. Bounded Model Checking

BMC was first proposed by Biere et al. in [10]. At the early days, BMC is based on SAT solving [11]. It is commonly acknowledged as an complementary technique to BDD based symbolic model checking [5]. Recent years, with the development of modern efficient SMT solvers like Z3 [12] and CVC4 [13] etc., there is a trend to use SMT solvers instead of SAT solvers in BMC for better expressiveness. The basic idea of BMC is to search for counterexamples (i.e., design bugs) in transitions (state space) whose length is restricted by an integer bound k. If no bug is found, then k is increased by one and the procedure repeats until either a counterexample is found or the pre-defined upper bound is reached.

B. Satisfiability Modulo Theories

SMT is a research topic that concerns with the satisfiability of formulas with respect to some background theories [14]. The development of SMT can be traced back to early work in the late 1970s and early 1980s. In the past two decades, SMT solvers have been well researched in both academic and industry, and achieved significant improving on performance and capability. Therefore, it has become possible to use SMT solver in BMC problem solving.

SMT is an extension of propositional satisfiability (SAT), which is the most well-known constraint-satisfaction problem [9]. SMT generalizes boolean satisfiability (SAT) by adding equality reasoning, arithmetic, fixed-size bit-vectors, arrays, quantifiers, and other useful first-order theories. An SMT solver is a tool for deciding the satisfiability (or dually the validity) of formulas in these theories [12]. In analogy with SAT, SMT procedures (whether they are decision procedures or not) are usually referred to as SMT solvers [15].

$$BMC(M, P, k) = I_0 \wedge \bigwedge_{i=0}^{k-1} T_i \wedge (\neg P) \qquad (1)$$

In BMC, states and transitions among them are encoded to logic formulas like (1). Then the encoded formula is sent to a SMT solver. Solving the formula (namely, SMT solving) involves computation-intensive processes and is thus time-consuming. Furthermore, as the model-checking bound increases, the encoded formulas become larger in size and harder to solve. In certain circumstances, the time in solving the formula may be unacceptably long. Therefore, an acceleration is needed for this procedure.

III. Previous Work

A. Overview

We have done some preliminary work in [1] on accelerating SMT solving procedure by using Message Passing Interface (MPI) [16], [17] and Open Multi-Processing (OpenMP) [18]. Our attempt is distributed SMT solving. MPI is used to implement distributed computing (i.e., multi-CPUs) and OpenMP is used for multi-cores parallel computing. As mentioned in Section I, there are two primary tasks in SMT-based BMC. Acceleration techniques applicable to either task can increase the whole solving efficiency. In our implementation, we choose to accelerating the second task – SMT solving procedure.

We have implemented distributed SMT solving in C language, using Z3 SMT solver for satisfiability verification. The system has a Client/Server (C/S) architecture. The topology of the network is shown in Figure 1. All clients are connected to a center server. Data is transmitted between server and clients. The server responds to requests for acquiring files from clients. If there exist enough SMT files, then the files will be sent to the target client. The SMT solving procedure happens on the clients after receiving SMT files from the server. OpenMP is used to create multiple threads, each thread invokes a Z3 SMT solver to solve specific SMT files. The solving procedure will be finished until the server has no file to send.

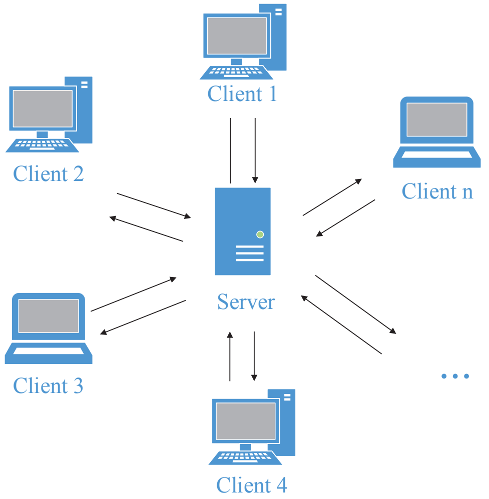


Figure 1: Network Topology

We conducted a series of experiments on six groups of benchmarks downloaded from the Satisfiability Modulo Theories Library (SMT-LIB) [19] that conform to version 2.0 of the SMT-LIB format. The benchmarks are `AUFNIRA, QF_UFLRA, AUFLIA, QF_UFLIA, QF_LRA-1`, and `QF_LRA-2`. The results shown in Table I and II are exciting. In Table I, the four benchmarks are easy problems, which means that SMT files can be solved in a short time. In addition, benchmarks in Table II are time consuming problems. The second column of the two tables shows the runtime which is obtained by applying sequential SMT solving. The third to fifth columns show the time of distributed SMT solving. The number of PCs (client) connected to the server is increased by one each time from 1 to 3 clients. The results in Table I show that the distributed solving strategy are proved effective for easy problems. We can obtain more than three times faster than serial solving in most benchmarks. The results, which are shown in Table II, are more positive when hard problems are considered. In the best case, the solving speed was raised by 36 times (3 clients are connected) comparing to the serial solving.

Obviously, in this way, not only the capacity/scalability, but also the solving speed of bounded model checking can be increased significantly. However, our strategy can not increase solving speed in all circumstances. When the problems to be solved are all small and easy solved, the efficiency boost is very limited. In the worst case, the speed only increased by two times even 3 clients are used.

TABLE I: MEASUREMENTS OF EASY PROBLEMS (SECOND)

Benchmarks	Serial	1 Client	2 Clients	3 Clients
QF_UFLRA	191.10	52.36	23.81	17.41
AUFNIRA	31.30	30.64	10.55	6.65
AUFLIA	105.80	88.87	41.90	50.48
QF_UFLIA	114.52	72.58	31.65	25.84

TABLE II: MEASUREMENTS OF HARD PROBLEMS (SECOND)

Benchmarks	Serial	1 Client	2 Clients	3 Clients
QF_LRA1	2465.78	1976.71	470.93	366.55
QF_LRA2	14796.03	11265.31	558.64	409.28

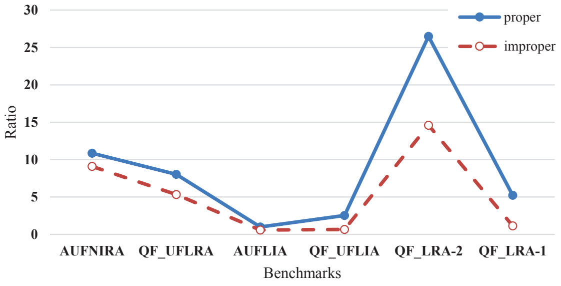


Figure 2: Comparison of Proper/Improper Task Dispatch

TABLE III: MEANING OF THE SIGNAL `power[0]`

`power[0]`	Meanings
`0`	Request files from a client or server has files to be send.
1	This client will be terminated.
2	No file in the server.

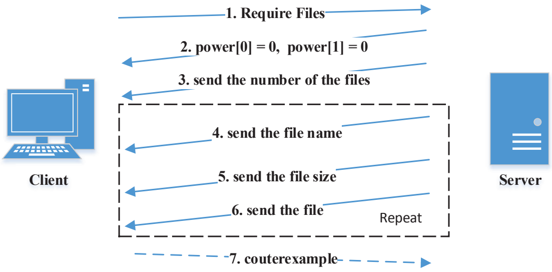


Figure 3: File Transferring Protocol

B. Shortcoming

Although our attempt gains significant improvement on solving performance, there are still some shortcomings of our previous work. The first is load balance, which is the core problems in the development of distributed model checker [20]. It means that our previous file distribution strategy is inefficient for scenarios where the hard problems or the combination of the easy and hard problems are considered. By *hard problems*, we mean problems that consume, e.g., 600 seconds or more per file in our experiment. The easy problems often take less than 1 second per file. In our previous work, the files are sent to clients by group. That means 4 files as a group are sent to a client after one file request. The server chooses files randomly. In other words, a client may receive easy problems as well as hard problems. For instance, there are four tasks named `Task1, Task2, Task3` and `Task4`. `Task4` is a hard problem and takes more time to solve. In a client, the 4 tasks are solved in parallel on different CPU cores. After `Task1 - Task3` are finished, `Task4` is still being handled. In this case, 3 CPU cores are idle and no new tasks is assigned to them until `Task4` is finished. The best case is that all files have the same solving hardness. The more different the computing divergences are, the longer the total solving time is. A primitive experiment has been done to demonstrate this shortcoming. The result is shown in Figure 2. The two lines denote time-improvement ratio of distributed solving to serial solving. The blue solid line denotes the results where the workload is dispatched evenly to all clients (called *proper* case) while the red dash line denotes uneven dispatching (called *improper* case). It is clear that the proper dispatching gains higher improvement ratio than the improper case. It should be noted that this experiment is a trivial one just for demonstrating the importance of task dispatch. To summarize, an improper task dispatching can slow down the whole solving procedure.

The second shortcoming is that there are some unnecessary communication between the server and clients during file transfer. In our previously proposed file transferring protocol, which is shown in Figure 3, when we try to transfer one file to a client, a 3-time communication is needed (step 4 to step 6). Actually it is unnecessary to send two messages in step 3 and step 4. If we did so, we have to spend more time on establishing connections. In Table I (recall that all tasks in this table are easy ones that can be solved less than 1 second), when we increase the number of clients, the improvement are limited. One of the reasons is that the delays brought by establishing connections and the data transmission overwhelm the superiority gained by distributed computing. In addition, not only the file transferring stage but also other unnecessary communication between the server and clients could be reduced. In Figure 3, the array `power[]` is used to send controlling signals. The first element `power[0]` stores the signal's type. The second element `power[1]` is used to indicate the source of a message. The values and the meaning represented by the value are shown in Table III.

IV. Improvement on Previous Distributed SMT Solving

The purpose of our distributed SMT solving is increasing the solving performance by leveraging computing resources of multiple computers as much as possible. In Section III, we have discussed two main shortcomings in our previous work. These shortcomings prevent our distributed implementation from further enhancing the performance of the distributed solving. We try to improve the utilization of computing resources in the following two aspects.

A. Fine-grained Dispatching Scheme

The first considered aspect is changing file dispatching grain from coarse-grain to fine-grain. Our previous file dispatching scheme is coarse-grained. That means the minimum unit to assign workload is client which realized by one MPI process even 4 threads running in it. The client (process) sends request to the server and receives returned files. After it receives files from the server, the client dispatches files to different threads where the SMT solving is done in a parallel way. This procedure is shown in Figure 4. In this figure, the part boxed by a dash rectangles denotes a client connected to the server. The whole procedure starts from sending file requirement from a client to the server for the first time. If there exits at least one file on the server, they will be sent to the client. Then the control flow enters the parallel solving stage. After the parallel solving, all four threads need to synchronize with each other before a new require-receiving round. The synchronization shown in Figure 4 means that threads which have finished their tasks in current round have to wait for other threads which are still in solving to finish their tasks. After the synchronization, the client will request new files again and the procedure will be repeated. The synchronization is needed because the file is dispatched to a client not thread. From the macroscopic view, the work load is dispatched to a client on process-level and the synchronization of threads will cause the shortcoming we discussed in the above section.

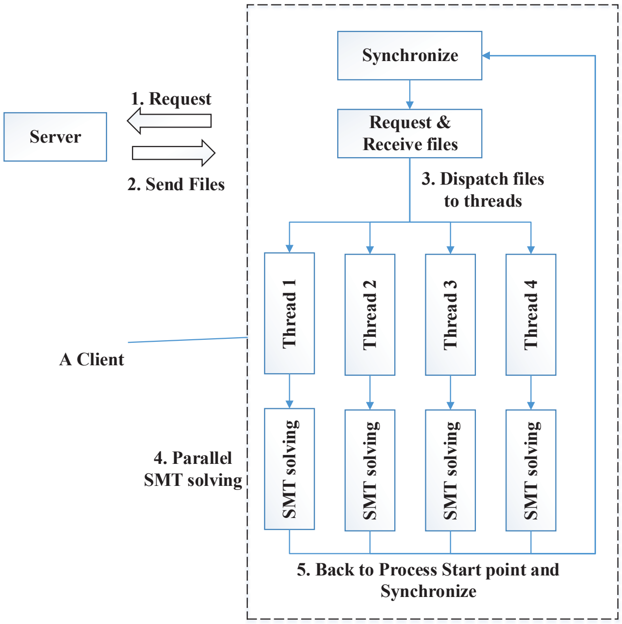


Figure 4: Previous File Dispatching Scheme

We change the dispatching scheme described above so that the synchronization between threads is removed after finishing one solving round. The improved scheme is shown in Figure 5. In our new scheme, the client starts from sending initial request to the server and receiving the first file set. It should be noted that this requesting-receiving round is executed only once. Then the received files are solved by 4 threads respectively in parallel. After that, 4 threads (in one client) will send file requests to the server separately when they need new files. The following receiving procedure is conducted by these threads also. If new files were received, threads will enter parallel SMT solving procedure again until a counterexample is founded or no files in the server. All four threads conduct the same procedures so that we admitted the detail of Thread 2 to Thread 3 in Figure 5. At this point, no synchronization is needed. Threads in a client are more independent than the previous scheme. The function of requiring and receiving files is implemented in thread level. Each thread can obtain new tasks from the server by itself. It is no longer necessary to wait for other threads to finish their solving tasks.

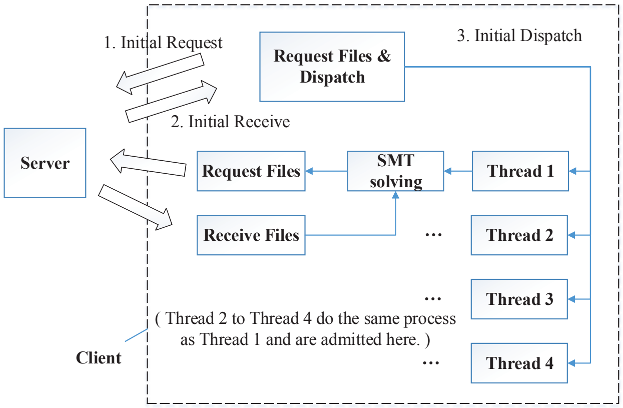


Figure 5: New File Dispatching Scheme

Algorithm 1. Newly Designed Client Procedure

```
 1. Process_n (int process_id) {
 2.   Initialization and definition of variables;
 3.   MPI_Send (request files, to process_0);
 4.   MPI_Recv (file_existence_condition from server);
 5.   if (no file is founded)
 6.     return 0;
 7.   j = 0;
 8.   while (j < file_num) {
 9.     MPI_Recv(file_information from server);
10.     MPI_Recv(file from server);
11.     fwrite (file to local HDD);
12.     rename(file);
13.     Clear receiving buffers;
14.     j++;
15.   }
16.   invoke parallel_solving(char *working_path);
17.   clean local files;
18.   MPI_Send(finish_signal to server);
19.   return 0;
20. }
```

In our implementation with the new designed fine-grained dispatching scheme, the architecture is still C/S. The server runs in a loop to receive the messages sent by the clients and prepare files for them. So we only expand the length of the control message `power[]` without any other changes. A major change comes up on the client side so we present it here in Algorithm 1. The argument `process_id` is an unique `int` number to distinguish a process which is running as a client. At the beginning, the client sends a request to the server (Line 3) and receives the file existence condition. The function `MPI_Send()` is a message sending function supplied by MPI [16]. It is a basic blocking message send operation. Routine returns only after the application buffer in the sending task is free for reuse. The function `MPI_Recv()`, which also is a MPI supplied function, receives a message and block until the requested data is available in the application buffer in the receiving task. The two functions must be used in as a pair. Otherwise, a dead block will happen. If file exists, an initial receiving and dispatching procedure will be done (Line 8 to Line 15). The initial dispatching is done by a client in our design because at the beginning, all CPU cores are idle, there is no need to consider the load balance problem at that time. The parallel SMT solving will take place by invoking the function `parallel_solving()` with the parameter `char *working_path` which indicates the local path where the target files are saved in.

The parallel solving part is also changed. It is done inside each client. We use OpenMP [18] to create 4 threads to leverage computing capacity of clients as much as possible. As we mentioned above every thread in a client now has the

Algorithm 2. Newly designed Function parallel_solving()

```
parallel_solving (char *working_path) {
  omp_set_num_threads(m);
#pragma omp parallel
  {
      switch (omp_get_threadnum()) {
      case 0:
      {
        Exploring all files in working_path {
          invoke Z3 to solve;
          goto next file;
        }
        do{
            Require and Receive files from the server;
            if (no file is founded)
              break;
            receive all files in this round;
            Exploring all files in working_path_t0 {
              invoke Z3 to solve;
              goto next file }
            delete solved files in working_path_t0;
        } while(server_has_files)
      }
        break;
    ... // Case 1 to case 3 are mostly like case 0 and omitted here
      }
  }
  return 0;
}
```

ability to request files from the server if necessary. Meanwhile, the files sent by the server will be transferred to the thread to achieve our fine-grained dispatching. We show the procedure of function `parallel_solving()` in Algorithm 2. In each client, firstly, a thread resolves the files dispatched in the initial step of the client. Then it enters a `do {...} while()` loop to request and receive files until there is no file in the server. After one receiving round, which means that a thread has received a set of files from the server, the received files are solved by Z3 SMT solver. The number of files in a set is a variable and its value is set to 2 by default. The argument `working_path_t0` is generated from the argument `working_path` to indicate its own working path. Thread 0 to Thread 3 are running in parallel and we omit the pseudo-code of thread 1 to thread 3 in Algorithm 2. If a counterexample is found in a solving round, the thread will report to the server. This activity of a counterexample finding and reporting is the same as our previous one so that we omit it in this algorithm.

In practical implementation, the new scheme might be less effective for the case which the solved problems are all easy problem or easy problem dominated (most of files are easy problems). In such case, the solving time of a single file is short enough. In our previous scheme, the main time consuming is the synchronization of threads. Even if the previous scheme is used, the synchronization time is a short duration. It will not effect the total solving time using the synchronization or not. However, by using the new dispatching scheme, we expect to get better performance for solving hard problems or combined problems under the server's random file dispatching strategy.

B. Communication Reduction

The second aspect to improve solving efficiency is by reducing unnecessary communications between the server and clients. In Figure 3, we can emerge step 2 with step 3 firstly. We expand the control signal `power[]`, which is an array with two elements, to 3 elements. For instance, if the `power[0] = 0` (means there exists files on the server), `power[2]` will be the number of the files while `power[1]` denotes the source of the message. If not, `power[2]` will set to 0 and `power[0] = 2`. In the file sending stage, before sending file data, the server will informs the file names and sizes, which are used by the client to create a receiving buffer dynamically. We use a `struct`, which consists of the name and size of the file to be sent in the future, and the structure could be sent by using `MPI_send()` function once. The step 4 and step 5 in Figure 3 can be merged by using the new data structure.

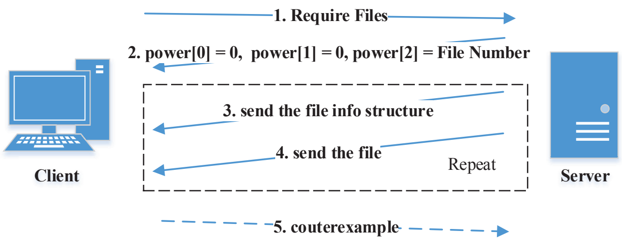

1. Require Files
2. power[0] = 0, power[1] = 0, power[2] = File Number
3. send the file info structure
4. send the file
Repeat
Client
Server
5. couterexample

Figure 6: New File Transferring Protocol

The reformed protocol is shown in Figure 6. The new File transferring protocol needs one communication to send the control information and two communications before sending one single file, while the previous protocol needs two and three communications, respectively. The first merging brings an expansion of the array size from 2 to 3. In C language, one `int` element consumes 2 Byte memory. For a modern PC and Ethernet, 2 Byte is not an issue. In the second merging we use a `struct` to store the file name and size. Comparing with sending these information separately, it consumes more bandwidth for one time sending. Even so, the increased bandwidth overhead is noting to a 100M/1000M Ethernet.

$$T_{pre} = n*(\bar{s}+t_{name}+t_{size})+(t_{control}+t_{number})*n/number \quad (2)$$

$$T_{new} = n*(\bar{s}+t_{structure})+t_{control}*n/number \quad (3)$$

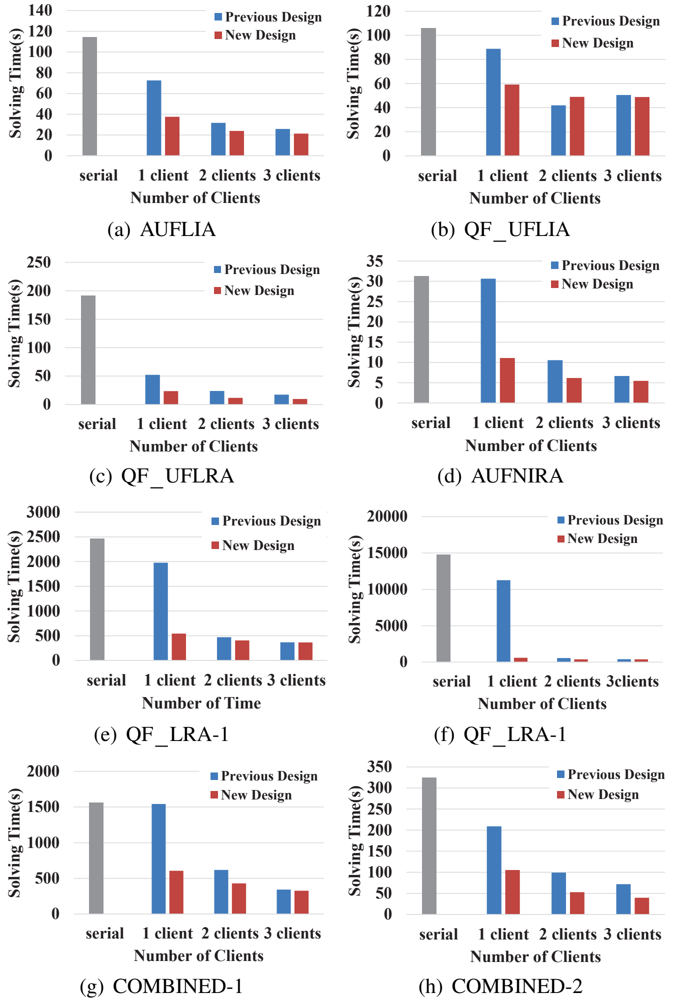
However, not every case can gain positive effect on promoting solving performance. If the target problems are all hard problems the communication overhead is not obvious comparing with the solving time. But for easy problems, the situation is opposite. The solving time of a easy problem is much more shorter than establishing communication and transfer control messages. The constitution of previous distributed SMT solving time T_{pre} is shown in 2. n denotes the total number of files to be solved, $\bar{s}$ is the average solving of a single file. t_{name} and t_{size} represent the time of establishing communication and sending file's name and size respectively. $t_{control}+t_{number}$ are consumption of sending control signal and the number of files. $number$ denotes files which will be sent by the server over one requirement of a client. After the reduction, the time consumption constitution T_{new} is shown in 3. In some cases, the average solving time $\bar{s}$ is no match for establishing the connection between the server and client. So reducing the time denoted by t_x can increase the performance significantly. t_x presents the time consummation of t_{name}, $t_{control}$ and t_{number}.

V. Experiments and Analysis

To evaluate the efficiency of our improved distributed SMT solving implementation, we conduct a serial experiments on six groups of benchmarks downloaded from the Satisfiability Modulo Theories Library (SMT-LIB) [19] that conform to version 2.0 of the SMT-LIB format. The benchmarks are same as our previous work which are `AUFNIRA, QF_UFLRA, AUFLIA, QF_UFLIA, QF_LRA-1`, and `QF_LRA-2`. We will not go into the details of those benchmarks, which are basically not relevant to the topic of this paper. Please refer to [19] for the meaning of those benchmarks. We deploy the program of the above described algorithms in four PCs running Windows 7 Enterprise Edition with MPICH 2.0 installed. One of the PCs is used as the server and the others are as clients. The hardware of the PCs are as follows: PC1 has a quad-core Intel Xeon CPU (2.66GHz) with 8GB RAM; PC2 and PC3 have a quad-core Intel i7 CPU (2.7GHz) with 8GB RAM; PC4 has a Intel Core2 Duo dual-core CPU (1.8GHz) with 2GB RAM. All the four PCs are connected to 100MB Ethernet. To evaluate the effective of our improvement, we use our prototype to solve those benchmarks and compare the results of our previous work. As we mentioned in the previous section, the four benchmarks, which are `AUFNIRA, QF_UFLRA, AUFLIA` and `QF_UFLIA`, are easy problems and the benchmarks `QF_LRA-1` and `QF_LRA-2` are hard problem. We design some new experiments beside the previous one. `Combined-1` is a combination of easy problems with hard problems and `Combined-2` is a set of 3000 easy problems. We use our previous algorithm and the new algorithms on solving these benchmarks respectively to prove the effectiveness of the latter.

Firstly, we conduct same experiments using the improved implementation for benchmarks which are used in our previous experiments [1]. We perform a serial solving experiment for each benchmarks using PC1. The SMT files are solved one after another in a serial way. Then the clients are connected to the server one by one and the same benchmarks are solved. PC4 is used as the server. Secondly we perform the same procedure mentioned above on new benchmarks `Combined-1` and `Combined-2`. The results are shown in Figure 7. The vertical rectangular marked as grey denotes the solving time of serial solving. The blue vertical rectangular presents results using previous algorithm. The red vertical rectangular denotes the solving time by using our new algorithm with two improvements. It is obvious that our improvements are useful on accelerating our previous distributed SMT solving implementation, especially for combined benchmarks. In Figure 7(e) and 7(f) when we add clients to 2 and 3, the improvement seems elusive. The reason is that these two benchmarks contain one or more hard problems which take nearly 300 seconds to be solved. In other words, the limit of distributed solving time is about 300 seconds no matter how many clients are connected.

Our new improved architecture give us the ability to control the usage of CPU cores more precisely. This means that we can add threads involved in parallel solving procedure one by one, in an easier way than before. We conduct experiments with `Easy Benchmarks` and `Combined Benchmarks` to investigate the influence by increasing of CPU cores. We use PC4 as the server and other PC as clients. At first one client is connected, but only one CPU core is used, the second time the number of the CPU cores is increased to 2. Four cores will be used on each PC, after one PC reached the max value of used CPU cores, new client will be connected. We increase



(a) AUFLIA

(b) QF_UFLIA

(c) QF_UFLRA

(d) AUFNIRA

(e) QF_LRA-1

(f) QF_LRA-1

(g) COMBINED-1

(h) COMBINED-2

Figure 7: The Distributed SMT solving Results

the CPU cores by one each time and repeat this procedure with two benchmarks respectively. The results are shown in Figure 8 and Figure 9. The results show that the curve of solving speed decrease sharply until the fourth CPU cores are involved. After that, the curve becomes flat. We have mentioned the possible reason in the paragraph above. For solving the `Combined Benchmarks`, the solving time of the hardest single problem is a limit of distributed SMT solving. For `Easy Benchmarks`, due to the CPU cores are on different PCs which are distributed in networks, the more CPU cores involved, the more communication will take place. Considering the time consumption resolving single easy problem and the overhead taken by network communication, the whole communication time consumption will be the predominant factor. In other words, if the solving target is determined, the whole solving time consumption could not be decreased always by simply adding more clients.

VI. Conclusion

In this paper, we first described our previous work on accelerating SMT solving using a distributed computation architecture, and discussed its shortcomings. To tackle those shortcomings, we proposed the fine-grained dispatching scheme and communication reduction methods. A series of experiments

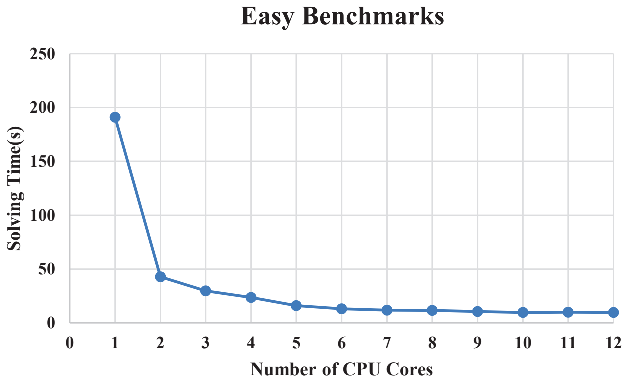


Figure 8: Influence by increasing CPU cores on easy benchmarks

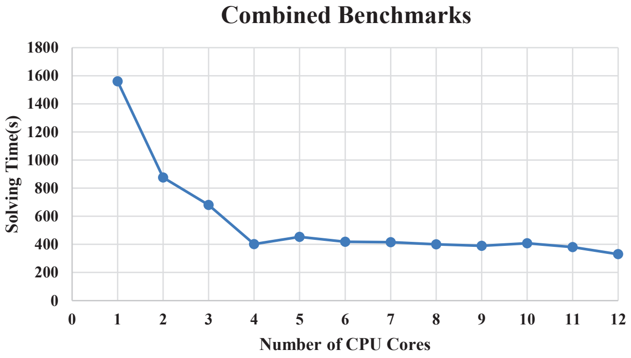


Figure 9: Influence by increasing CPU cores on combined benchmarks

were carried out to demonstrate the feasibility and efficiency of our improved techniques. Discussions and analysis are raised after the experiments.

Regarding future work, in addition to the techniques methods proposed in this paper, there are other methods that can be used for improving the efficiency of distributed SMT solving. The methods proposed in this paper are only for the client side. However, we can actually further improve the efficiency from the server side as well. In our current implementation, the number of requests from clients is four times higher than before, which may make the server get stuck. The server responds to the clients' requests in a serial way while parallel I/O can be used to give the server an ability to respond to various requests at the same time. Currently, the server randomly chooses files to send to the clients without considering the computation ability of different clients. Another possible idea is that the sever could use other optimized file choosing strategy, e.g., by the size of files, so as to avoid dispatch hard problems to weak clients. We will investigate those possibilities in the future.

References

[1] L. Liu, W. Kong, and A. Fukuda, "Implementation and Experiments of a Distributed SMT Solving Environment," International Journal on Computer Science and Engineering, vol. 6, 2014, pp. 80–90, ISSN: 0975-3397.

[2] E. M. Clarke, O. Grumberg, and D. Peled, Eds., Model Checking. The MIT Press, 1999, ISBN: 978-0-262-03270-4.

[3] G. J. Holzmann, Ed., The SPIN Model Checker: Primer and Reference Manual. ADDISON WESLEY Publishing Company Incorporated, 2003, ISBN: 978-0-321-77371-5.

[4] J. R. Burch, E. M. Clarke, K. L. McMillan, D. L. Dill, and L.-J. Hwang, "Symbolic model checking: 10^{20} states and beyond," Information and Computation, vol. 98, no. 2, 1992, pp. 142–170.

[5] A. Biere, A. Cimatti, E. M. Clarke, O. Strichman, and Y. Zhu, "Bounded Model Checking," Advances in computers, vol. 58, May 2003, pp. 117–148.

[6] A. Armando, J. Mantovani, and L. Platania, "Bounded model checking of software using SMT solvers instead of SAT solvers," International Journal on Software Tools for Technology Transfer, vol. 11, no. 1, Nov. 2008, pp. 69–83.

[7] N. Amla, R. Kurshan, K. L. McMillan, and R. Medel, "Experimental analysis of different techniques for bounded model checking," in Tools and Algorithms for the Construction and Analysis of Systems. Springer, 2003, pp. 34–48.

[8] L. de Moura and N. Bjørner, "Satisfiability modulo theories: An appetizer," in Formal Methods: Foundations and Applications. Springer, 2009, pp. 23–36.

[9] L. De Moura and N. Bjørner, "Satisfiability modulo theories: introduction and applications," Communications of the ACM, vol. 54, no. 9, 2011, pp. 69–77.

[10] A. Biere, A. Cimatti, E. Clarke, and Y. Zhu, "Symbolic Model Checking without BDDs," in In Proc. of the Workshop on Tools and Algorithms for the Constrction and Analysis of Systems (TACAS'99). Springer Berlin Heidelberg, 1999, pp. 193–207.

[11] C. Barrett, R. Sebastiani, S. Seshia, and C. Tinelli, Handbook of Satisfiability. IOS Press, 2009, vol. 185, ch. 26, pp. 825–885.

[12] L. de Moura and N. Bjørner, "Z3: An efficient SMT solver," in Tools and Algorithms for the Construction and Analysis of Systems. Springer, 2008, pp. 337–340.

[13] "CVC4: the SMT Solver," 2014, URL: http://cvc4.cs.nyu.edu/web/ [accessed: 2014-01-18].

[14] A. Biere, Handbook of satisfiability. IOS Press, 2009, vol. 185.

[15] C. Barrett, R. Sebastiani, S. A. Seshia, and C. Tinelli, "Satisfiability modulo theories," in Handbook of Satisfiability. IOS Press, 2009, pp. 825–885.

[16] "The Message Passing Interface (MPI) Standard," 2014, URL: http://www.mcs.anl.gov/research/projects/mpi/ [accessed: 2014-01-02].

[17] "MPICH User's Guide (Version 3.0.4)," 2014, URL: http://www.mpich.org/static/downloads/3.0.4/mpich-3.0.4-userguide.pdf [accessed: 2014-01-02].

[18] "Open MPI: Open Source High Performance Computing," 2014, URL: http://openmp.org/wp/2013/12/tutorial-introduction-to-openmp/ [accessed: 2014-01-02].

[19] "SMT-LIB: The Satisfiability Modulo Theories Library," 2013, URL: http://www.smtlib.org/ [accessed: 2013-12-10].

[20] G. J. Holzmann and D. Bosnacki, "The Design of A Multi-core Extension of the SPIN Model Checker," IEEE Trans on Software Engineering, vol. 33, no. 10, 2007, pp. 659–674.

Merchant Facial Expressions and Customer Trust in Virtual Shopping Environment

Nasser Nassiri
Department of Information Technology
Higher Colleges of Technology (HCT)
Dubai, UAE
nasser.nassiri@hct.ac.ae

David Moore
Department of Computer Science
Leeds Metropolitan University (ex professor)
Leeds, UK
moore-exleedsmet@hotmail.co.uk

***Abstract* - Trust is an essential contributor to the traditional customer experience. Online, it is harder for individuals to assess a partner's trustworthiness, as many of the cues present in face-to-face interaction are difficult to transmit via technology. The recent advance of computer graphics and internet technology, however, potentially enables individuals to transmit these cues through their avatars in 3D e-commerce environments. The paper investigates the impact of the vendor avatar's dynamic facial expression on consumer trust in 3D e-commerce environments. An experiment was conducted to empirically test the effects of the basic seven universal emotions of facial expression displayed by online vendor avatars on consumer trust in a 3D e-commerce environment. Respondents were able to recognize the intended emotions on the salesman avatars. They preferred to purchase from the salesman with a neutral expression.**

Keywords-Virtual Shopping Environments; Trust; Avatars; Online Retailing; Virtual Salespersons

I. INTRODUCTION

The Internet has brought customers and retailers together by enabling consumers to shop with anyone, at any time and from anywhere. However, it is also keeping them apart, and consumers are no longer in face-to-face contact with salespeople. Trust is an essential contributor to the traditional customer experience [11]. It makes collaboration between vendors and consumers more pleasant and allows for cooperation that would otherwise not take place. In online transactions, trust becomes even more important. Commercial interactions that are carried out over the internet carry more risk than face-to-face interactions [18]. One reason for this is that it is harder for individuals to assess a partner's trustworthiness, as many of the cues present in face-to-face interaction (e.g., emotions, posture, eye gaze, and gesture) were not transmitted via the early technologies used [11]. Yet, up to 93% of the meaning of all conversations comes from non-verbal communication [28]. However, the recent advance of computer graphics and internet technology enables individuals to transmit these cues through their avatars in Collaborative Virtual Environments (CVE) and therefore in 3D e-commerce environments such as that featured in Fig. 1 [5][9]. Avatars are virtual objects representing a human being in a virtual environment. These virtual objects are three-dimensional animated models. They are used by, for example, SecondLife [37], which offers retailers and shoppers a virtual 3D environment where both shoppers and retailers interact via their avatars.

Fabri et al. [16] conducted an experiment to establish how facial expressions of emotions might be effectively and efficiently captured and represented in CVE. Their findings provide strong evidence that creating virtual face representations with a limited number of facial features allows emotions to be effectively portrayed visually and gives rise to recognition rates that are comparable with those of corresponding raises a research question about whether the effectiveness of these avatars in conveying emotions will enable such avatars to transmit signals of trustworthiness in CVE e-commerce interactions" [23].

This paper, therefore, focuses on the role which the dynamic facial expressions of virtual vendors might play in 3D e-commerce environments. Specifically, it investigates the impact of the vendor avatar's dynamic facial expression on consumer trust in 3D e-commerce environments. The study reports an experiment in which participants were presented with seven avatars, each animated with one of the seven universal emotions, and saying the same message but with a varied tone of voice, to reflect the emotion shown by each avatar.

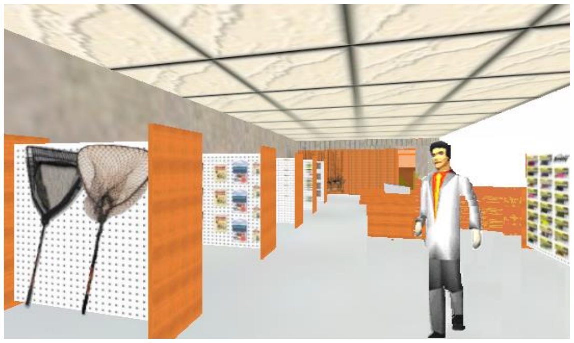


Figure 1. A simulation of a 3D e-commerce environment

II. THEORETICAL FRAMEWORK

Trust plays an important role in many social and commercial interactions. It is seen as a concept with many dimensions [1][8] and has been studied in many diverse disciplines. For instance, economists have focused on merchants' reputations and their effect on transactions [4]; researchers of marketing have focused on strategies for building consumers' trust [11][25][30]; human computer interaction scholars have focused on the relation between design and usability of a system and users' reactions [8]; and psychologists have studied trust as an interpersonal and group phenomenon [36].

Although trust is an essential component of all successful face-to-face commercial transactions, there is a renewed focus on trust when commercial transactions are conducted through electronic media and customer trust in internet-mediated marketing environments has been identified as a major research area [35]. Buyers look for signs from sellers that increase their trust, and sellers look at ways they can help build buyers' trust. Since uncertainties exist in transactions over the Internet, many researchers have stated that trust is a critical factor influencing the successful proliferation of e-commerce [19][20][21]. This has brought about new challenges for building trust in e-commerce business environments. Recent research on trust includes development of a trust typology [26], the measurement of trust [3], and critical factors influencing initial trust formation [26], including the impact of familiarity and seller size and reputation [20]. Mukherjee and Nath [31] have extended and validated Morgan and Hunt's [30] commitment-trust theory to online retailing contexts. Empirically, trust building has been identified as one of the main web experience components having a positive and significant effect on the selection of an e-vendor [24].

The initial impressions of an e-commerce website are particularly important. Although attitudes about a company or website evolve over time and even during the course of one shopping visit, initial impressions persist and can affect whether or not a visitor returns. Judgments about trustworthiness occur as soon as a visitor begins interacting with a site [6]. A number of researchers have investigated the importance of the website interface in promoting trust in the website from initial usage. Nielsen et al. [32] recommended that company information about pricing, including taxes and shipping costs, and balanced information about products would increase customers' perceptions of trustworthiness. However, besides cognitive elements, trust also includes an affective dimension, based on underlying feelings [25] and in service contexts, emotional bonds with customers have been found to provide a more enduring source of loyalty than economic incentives and switching costs [14]. Further, as Arnott [2] argues, decisions to trust an organization are based on assessments of several elements such as, in the case of online shopping experiences, trusting the brand, trusting the internet merchant and trusting the information system. Relationship marketing is more effective when the relationships are developed with individual people rather than with the firm itself [33]. In spite of this evidence that the human element is an important contributor to trust, many websites neglect this element and focus instead on securing trust though the use Socket Software Layer (SSL) and only a small number of these websites focus on the visual representation of the vendor's face to secure trust – a key reason for conducting this research.

The face is a very important source of socio-emotional cues. Centuries ago, Hippocrates advised doctors to use their facial expressions to establish a good rapport with their patients. Surakka and Hietanen [39] see facial expressions of emotion clearly dominating over vocal expressions of emotion, and Knapp [22] generally considers facial expressions as the primary site for communication of emotional states. Ekman et al. [12] found that, in addition to the neutral expression, there are six universal facial expressions, corresponding to the following emotions: Surprise, Anger, Fear, Happiness, Disgust, and Sadness (see Figure 2a). Fig. 2b shows the 7 universal emotions and neutral expression as represented in a virtual face corresponding to the real photos in Fig. 2a [16].

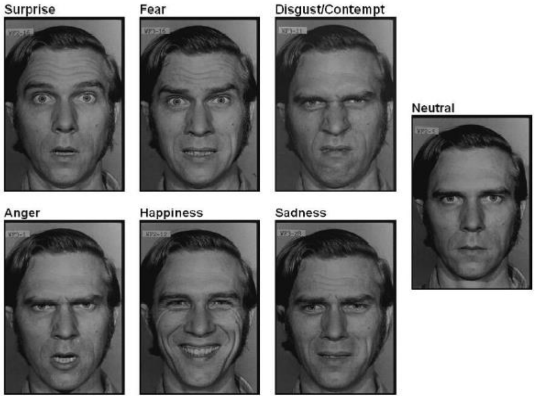


Figure 2a. Facial expressions of the b7 universal emotions [16]

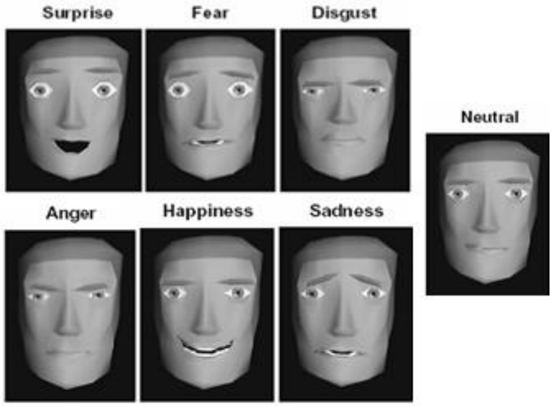
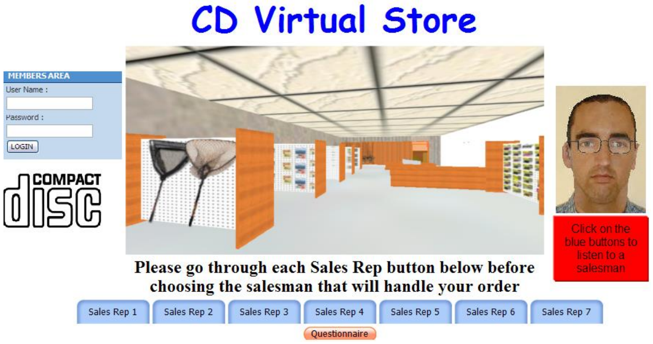


Figure 2b. The 7 universal emotions as represented in a virtual face [16]

Researchers have studied the impact of facial expression emotions on trust in face-to-face interactions. For example, in commercial transactions, trust building mechanisms include several factors such as physical presence, facial expression of emotions, and past action [18]. While this illustrates clearly the influence of facial expressions on trust in face-to-face interactions, little is known about its influence on trust in 3D e-commerce environments.

III. RESEARCH METHODOLOGY

Forty two Lebanese participants of different gender and ages were involved in the experiment. Lebanon is a particularly apt country in which to conduct the experiment since salesman-to-consumer interactions, bargaining, trust based on personally relationship have been extremely important in the traditional shopping environments of the Arab world [34]. Each participant was presented with a local website constructed for the purpose of this study (Fig. 3) with some pictures of CDs signifying that this site sells audio CDs online. The website includes seven buttons for each virtual vendor representing the seven facial expressions of emotions. Once the virtual face was created, a sound track was digitized, reflecting the tone of voice corresponding to the emotion represented by each facial expression. The sound was then attached to each facial expression. The facial expressions were then added to the animated talking virtual head manually using the same application (Fig. 4). Each participant was required to go through each animation by pressing its corresponding button. Upon pressing any button, the chosen virtual avatar face is animated for 10 seconds, showing the emotion chosen and saying, in a tone matching the emotion, a sentence encouraging the participant to place the order through him.



CD Virtual Store

Please go through each Sales Rep button below before choosing the salesman that will handle your order

Click on the blue buttons to listen to a salesman

Figure 3. The virtual website used for the experiment.

Each participant listened to the seven avatars and at the end of the session was asked to answer two questions. The first question required them to match each sales representative with the emotion he was perceived as conveying. The second question asked about the extent to which (using a Likert 6-point scale) the participant would be prepared to place an order with each of the sales representatives. The participants were free to visit the avatars in any order and were allowed to revisit them in the process of answering the questions. The gender and age-group of each participant were recorded, as well as their responses to the questions.

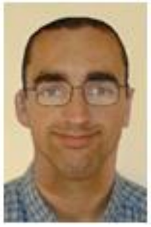


Figure 4.1.
Happy

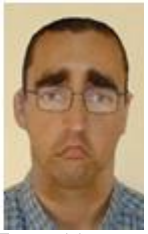


Figure 4.2.
Sad

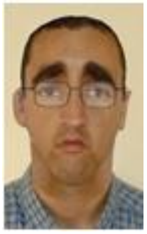


Figure 4.3.
Afraid

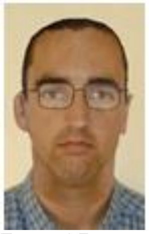


Figure 4.7.
Neutral

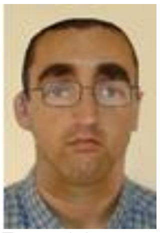


Figure 4.4
Surprised

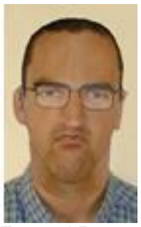


Figure 4.5.
Disgusted

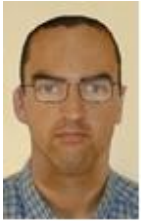


Figure 4.6.
Angry

IV. RESULTS

The gender and age group of the participants are shown in Table 1, which shows a reasonable representation across gender and age.

TABLE I. SUMMARY OF GENDER AND AGE GROUP OF PARTICIPANTS

Gender	Age Group (years)				Total
	20 to 25	26 to 30	31 to 40	Over 40	
Male	11	9	5	4	29
Female	9	2	0	2	13
Total	34	11	5	6	42

The first question was intended to investigate whether these particular instances of the universal emotion were correctly identified by the participants. The results of the participants' identifications are presented in Table 2.

All the participants successfully identified the happy, sad, angry and neutral expressions of the Sales Rep. There are a few misidentifications of the surprised, afraid and disgusted – 4, 7 and 9 misidentifications, respectively, with surprised being confused with afraid and disgusted, afraid being confused with disgusted and sad, and disgusted being confused with afraid, surprised and sad.

TABLE II : RECOGNITION OF THE SALES REPRESENTATIVE EMOTIONS

Intended Emotion	Perceived Emotion						
	Neutral	Happy	Sad	Afraid	Surprised	Disgusted	Angry
Neutral	42						
Happy		42					
Sad			42				
Afraid			2	35		5	
Surprised				2	38	2	
Disgusted			2	4	3	33	
Angry							42

To provide a measure of the strength of agreement between the intended emotion and the one perceived by the participants, Cohen's Kappa coefficient [7] of κ = 0.921, has been calculated. The coefficient represents the proportion of agreement after chance agreement is excluded and it takes negative values for agreement less than that expected by chance, a value of 0 for agreement levels expected by chance, and a value of 1 for perfect agreement. The value of the coefficient confirms the high rate of agreement between the emotion that the avatar is intended to convey and the emotion perceived by the participants.

It is interesting to note that only one of the female participants made any error in classification of the emotions, whereas 8 male participants made 19 errors between themselves. This may suggest that women are better at identifying the avatar's emotion, however, this was not found to be statistically significant ($\chi 2$ (df =1) = 2.110, p (exact) = 0.232, based on a two-by-two contingency table of the number of male and female participants identifying all the emotions correctly or making one or more mistakes).

The second question asked respondents whether they would place an order with each of the sales representatives. The results are described in Table 3.

The results in Table 3 show strong preference for the Neutral sales representative, followed by a slight inclination towards ordering with the happy sales representative. There is a slight disinclination to use the Surprised and Sad sales representatives, a disinclination to use the Afraid sales representative and strong disinclination to use the Disgusted and Angry sales representatives. In order to analyze these results a Repeat Measures General Linear Model [38] was constructed, with the emotion expressed by the sales representative as the within-subject variable and the gender and age of the participants as between-subject variables. Post-hoc comparisons were conducted on variables that proved to be significant within the model to identify differences. Mauchy's test for sphericity was not passed ($p < 0.001$), so the Greenhouse-Geisler correction [38] to the degrees of freedom of the univariate tests was applied, using $\varepsilon = 0.661$. The within-subject tests show that the emotion of the sales representatives with regard to placing an order is significant ($p < 0.001$), however the interactions of emotion with gender, age, or gender and age are not significant (p = 0.564, 0.600, and 0.151, respectively). Therefore, there are significant differences between the participants' responses to the emotions, but these responses are not significantly affected by the gender or age of the participants. Table 4 shows the estimated means and their 95% confidence intervals, ordered in decreasing preference. The responses were coded 'strongly disagree' = 1 to 'strongly agree' = 6.

TABLE III: LIKELIHOOD OF PARTICIPANTS PLACING AN ORDER WITH EACH SALES REPRESENTATIVE

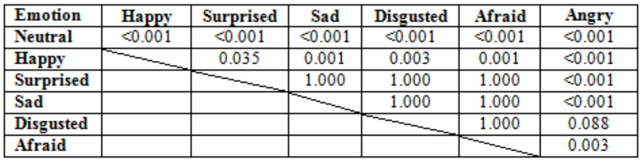
		1 **Strongly Disagree**	**2**	**3**	**4**	**5**	**6** **Strongly Agree**
Sales Rep2	**Neutral**				5	13	
Sales Rep5	**Happy**	2	7	5	21	4	3
Sales Rep7	**Surprised**	5	13	18	6		
Sales Rep6	**Sad**	8	12	18	4		
Sales Rep4	**Disgusted**	24	10	8			
Sales Rep1	**Afraid**	13	14	10		5	
Sales Rep3	**Angry**	38	4				

TABLE IV: MEAN, STANDARD ERROR AND 95% CONFIDENCE INTERVAL FOR REACTIONS TO SALES REPRESENTATIVE EMOTION, IN DESCENDING ORDER

	Emotion	**Mean**	**Standard Error**	**95% Confidence Interval**	
				Lower Bound	**Upper Bound**
Sales Rep2	**Neutral**	5.39	.14	5.11	5.67
Sales Rep5	**Happy**	3.66	.22	3.20	4.11
Sales Rep7	**Surprised**	2.37	.17	2.31	3.00
Sales Rep6	**Sad**	2.12	.17	2.02	2.73
Sales Rep4	**Disgusted**	2.11	.30	1.50	2.73
Sales Rep1	**Afraid**	1.14	.23	1.65	2.57
Sales Rep3	**Angry**	11	.06	1.02	1.25

Table 5 shows the results of the post-hoc comparisons between the participants' responses to the sales representatives' emotions. The Bonferroni correction [15] for multiple comparisons has been applied. Taking the results from Tables 4 and 5 together, the Neutral sales representative is very highly significantly preferred above the other emotions. Participants were neutral about the Happy sales representative, who is nevertheless significantly preferred to the Surprised sales representative. The participants were disinclined to place orders with the Surprised, Sad, Disgusted and Afraid sales representative and there were no significant differences between their reactions to these sales representatives. Participants were least likely to place an order with the Angry sales representative, whom they reacted to significantly less favorably than all the other sales representatives, with the exception of the Disgusted sales represented that was not significantly different.

TABLE V.POST-HOC COMPARISONS BETWEEN EMOTIONS (SIGNIFICANCE, P)



Emotion	**Happy**	**Surprised**	**Sad**	**Disgusted**	**Afraid**	**Angry**
Neutral	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
Happy		0.035	0.001	0.003	0.001	<0.001
Surprised			1.000	1.000	1.000	<0.001
Sad				1.000	1.000	<0.001
Disgusted					1.000	0.088
Afraid						0.003

V. DISCUSSION

This study considered whether the emotions conveyed through the animated facial expressions of avatars and their tone of voice could induce trust in 3D e-commerce environments. The results show that people successfully recognized the intended emotions conveyed by the animated salespeople, or avatars. In this respect, this study is in line with Fabri et al. [16] results, suggesting that human emotions are legible in 3D virtual environments. Further, it was interesting to find that while the results did not reach statistical significance, women made fewer mistakes in recognizing intended emotions than men. While further research is needed to validate this result, it might suggest that online as offline, women are better readers of emotions [29].

There appears to be a very high success rate for the recognition of the sales representative's emotion, albeit not 100% recognition. Where misidentification has occurred, it has been between related emotions (sad, afraid, surprised and disgusted) and not between more contrasting emotions: happy, neutral and sad.

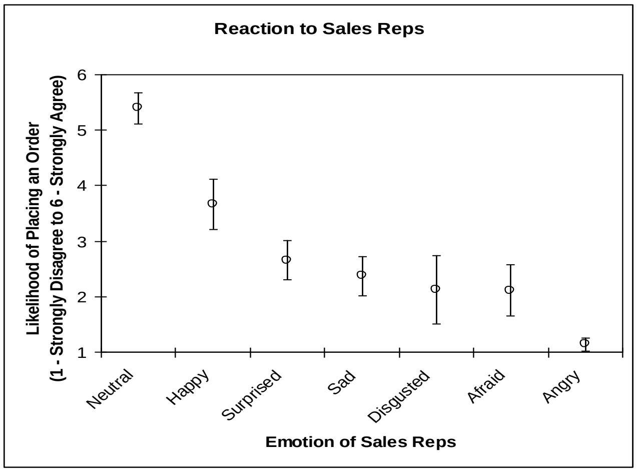


Figure 5. Mean and 95% Confidence Interval of the Participants' Inclination to use Sales Representative (1-Strongly Disagree to 6 - Strongly Agree)

The other set of results shows that respondents preferred to purchase from the salesman with a neutral expression, rather than from any other salesmen, including the happy salesman. This suggests that the perceived positive emotions of sales avatars do not induce trust as much as neutral expressions. This is perhaps surprising, since offline, the positive emotions of salespeople influence trust [18], and lead people to return to a store and spread word of mouth [40]. A possible explanation is that while consumers are able to read the emotions of sales avatars, they remain conscious of the fact they are not real people with real feelings.

It may be that rather than convincing them, positive emotions conveyed by an avatar make customers more 'suspicious' than a neutral expression. An alternative explanation of these results would be that the neutral expression in fact conveys seriousness, professionalism and competence. This would be consistent with Fogg et al. [17] study which mentions expertise as one of the elements inducing trust online.

Thus, consumers may develop trust as a result of perceptions of professionalism and competence, rather than emotionally-pleasant, virtual interactions. A similar finding emerged from a study of responsiveness during service interactions taking place over the telephone. Doucet [10] established that informational responsiveness were more beneficial for both the organization and the customer than emotional responsiveness. In a context precluding real face-to-face interactions, therefore, information and demonstration of competence may be the main drivers of trust.

There are several possible managerial implications of this study. First, it suggests that consumers are able to recognize intended emotions as conveyed by sales representative avatars. Therefore, the current technology makes it possible for online retailers to infuse their virtual sales people with certain emotions. Second, the participants' inclination to purchase from a neutral rather than a happy sales representative suggests that it may not be possible to increase purchase intentions purely by animating the virtual salesperson with a happy mood. It has been suggested that consumers may equate the neutral expression with competence, in which case retailers may be able to use further cues to convey competence, such as for example the sales representative's dress and language and of course domain knowledge. Further, it might be useful to include the effect of presenting "no avatar" in order to compare the overall feedback of the participants,

A limitation of the study is the relatively small sample size (n=42), in spite of the respectable number of observations (294). Future research should aim to investigate statistically whether women are better readers of emotions online than men.

VI. CONCLUSION AND AVENUES FOR FURTHER RESEARCH

This study suggests that contrary to traditional shopping environments where trust can be conveyed through the sales representative's facial expression, or the vocal expression of emotions, virtual environments do not seem able to increase customer trust through the positive emotions of sales representatives' avatars. Further research is needed to validate these initial results. It would be useful in particular to design an experiment which simulates two different interactions with a sales avatar, one based on social exchange, while the other is based on informational exchange. A qualitative approach, using think aloud [13] while people interact with different sales avatars, would also allow for a deeper understanding of consumers' reactions to virtual sales people, and of how virtual sales people can convey expertise.

Another limitation is that the same virtual salesman avatar is used to show all the facial expressions, and this might confuse the participants when scoring the expressions. Further research should also aim to have a different avatar for each expression. Future research should aim to verify statistically the possibility that women are better readers of emotions online than men. Further, the experiment was presented to the participants with avatars having different emotion and appropriate ton of voice. It was suspicious whether the voice contributed to the high success rate of the recognition of the sales representative's emotion or not.

Therefore, a future research in this direction might lead to more robust results.

We must reiterate that this study's results do not necessarily imply that the positive emotions of sales avatars are detrimental to 3D e-commerce environments. In particular, further research should establish whether the emotions of sales avatars make the shopping experience more 'fun', keep people in the virtual environment longer, and make them explore more. This in turn may commit consumers to return to the store, and become life-long customers.

By suggesting a preference of consumers for neutral rather than positive emotions on sales avatars' facial expressions, this study has contributed towards a better understanding of how consumers react to virtual interactions, and possible explanations as to why they prefer neutral expressions on sales representatives' avatars. As such, the study supports the argument that retailers face a number of new challenges in managing online customer experiences. The virtually of human interaction and its consequences is an important area for future research.

REFERENCES

[1] Araujo, A. (2003), "Developing trust in internet commerce", in Proceedings of the Conference of bthe Centre for Advanced Studies on Collaborative Research.

[2] Arnott, D.C. (2007), "Trust - Current Thinking and Future Research", European Journal of Marketing, vol. 2007 no. 41, pp. 981-87.

[3] Bhattacherjee, A. (2002), "Individual Trust in Online Firms: Scale Development and Initial Test", Journal of Management Information Systems, vol. 19, no. 1 (Summer), pp. 211-241.

[4] Cabral, B. (2006), "The Economics of Trust and Reputation: A Primer", Technical Report, New York University and CEPR.

[5] Capin, T., Pandzic, I., Thalmann, N., Thalmann, D. (1999), "Realistic Avatars and Autonomous Virtual Humans in VLNET Networked Virtual Environments", in Earnshaw, R.A. and Vince, J. (Ed.), Virtual Worlds on the Internet, IEEE Computer Science Press, pp. 157-174.

[6] Cherny, L. (1999), Conversation and Community: Chat in a Virtual World, CSLI, Stanford.

[7] Cohen, J. (1960), "A Coefficient of Agreement for Nominal Scales," Educational and Psychological Measurement, vol. 20 no. 1, pp. 37-46.

[8] Corritore, L. Kracher, B., and Wiedenbeck S. (2003), "On-line trust: concepts, evolving themes, a model", International Journal of Human-Computer Studies, vol. 58 no. 6, pp. 737–758.

[9] Coulson, M., (2002), "Expressing emotion through body movement: A component process approach", in Proceedings of AISB Symposium on Animated Expressive Characters for Social Interaction, London, UK, pp. 11-16.

[10] Doucet, L. (2007), Responsiveness: Emotion and Information Dynamics in Dyadic Service Interactions, PhD dissertation, University of Pennsylvania.

[11] Dwyer, F.R., Schurr, P.H., and Oh, S. (1987), "Developing Buyer-Seller Relationships", Journal of Marketing, Vvl. 51 no. 2, pp. 11-27.

[12] Ekman, P., Friesen, W., and Ellsworth, P. (1972), Emotion in the Human Face: Guidelines for Research and an Integration of Findings, Pergamon Press Inc, New York, NY.

[13] Ericsson, A and Herbert, S. (1993), Protocol Analysis - Verbal Reports as Data, The MIT Press, Cambridge, MA.

[14] Evanschitzky, H., Iyer, G.R., Passman, H., Niessing, J., and Meffert, H. (2006), "The Relative Strength of Affective Commitment in Securing Loyalty in Service Relationships," Journal of Business Research, vol. 59 no. 12, pp. 1207-13.

[15] Everitt, B. (1995), The Cambridge Dictionary of Statistics in the Medical Sciences, Cambridge University Press, Melbourne.

[16] Fabri, M., Moore, D., and Hobbs, D. (2002), "Expressive Agents: Non-verbal Communication in Collaborative Virtual Environments", in Proceedings of Autonomous Agents and Multi-Agent Systems 2002 (Embodied Conversational Agents Workshop), July 2002, Bologna, Italy.

[17] Fogg, B.J., Marshall, J., Laraki, O., Osipovich, A., Varma, C., Fang, N., Paul, J., Rangnekar, A., Shon, J., Swani, P., and Treinen, M. (2001), "What makes Web sites credible? A report on a large quantitative study", CHI 2001 Conference Proceedings, vol. 3 no. 1, pp. 61-6.

[18] Giddens, A., (1990), The Consequences of Modernity, Stanford University Press, Stanford.

[19] Hoffman, L., Novak, P., and Peralta, M. (1999), "Building Consumer Trust Online", Communications of the ACM, vol. 42, no. 4, pp. 80-85.

[20] Jarvenpaa, L., Tractinsky, J., and Vitale, M. (2000), "Consumer trust in an internet store", Information Technology and Management, Vol. 1 No. 1 and 2, pp. 45-71.

[21] Keen, W. (2000), "Ensuring e-trust", Computer World, vol. 34 no. 11, pp. 46.

[22] Knapp, M. (1978), Nonverbal Communication in Human Interaction (2nd Edition), Holt, Rinehart and Winston Inc., New York, NY.

[23] Lisetti, C., Nasoz, F, Lerouge, C., Ozyer, O., and Alvarez, K. (2003), "Developing Multimodal Intelligent Affective Interfaces for Tele-Home Health Care", International Journal of Human Computer Studies Special Issue on Applications of Affective Computing in Human-Computer Interaction, vol. 59 no. 1-2, pp. 245-255.

[24] Lorenzo, C., Constantinides, E., Geurts, P., and Gomez, M.A. (2007), "Impact of Web Experience on E-Consumer Responses", in 8th International Conference on E-Commerce and Web Technologies, ed. G. Psaila and R. Wagner, Regensburg, Germany.

[25] McAllister, D.J. (1995), "Affect- and Cognition-Based Trust as Foundations for Interpersonal Cooperation in Organizations", Academy of Management Journal, vol. 38 no. 1, pp. 24-59.

[26] McKnight, D. and Chervany, N. (2001), "What Trust Means in E-Commerce Customer Relationships: An Interdisciplinary Conceptual Typology", International Journal of Electronic Commerce, vol. 6 no. 2, pp. 35-59.

[27] McKnight, D., Cummings, L., and Chervany, N. (1998), "Initial Trust Formation in New Organizational Relationships", Academy of Management Review, vol. 23 no. 3, pp. 473-490.

[28] Mehrabian, A. (1971), Silent Messages, Wadsworth, Belmont, CA.

[29] Merten, J. (2005), "Culture, Gender and the Recognition of the Basic Emotions", Psychologia, vol. 48 no. 4, pp. 306-316.

[30] Morgan, R. M. and Hunt, S. D. (1994), "The Commitment-Trust Theory of Relationship Marketing", Journal of Marketing, vol. 58, pp. 20-38.

[31] Mukherjee, A. and Nath, P. (2007), "Role of Electronic Trust in Online Retailing - a Re-Examination of the Commitment-Trust Theory", European Journal of Marketing, vol. 41, pp. 1173-202.

[32] Nielsen, J., Molich, R., Snyder, C., and Farrell, S. (2000), "E-commerce User Experience", Technical Report, Nielsen Norman Group, Cheskin Research & Studio Archetype/Sapient: 1999, eCommerce Trust Study, Sapient, http://www.sapient.com/cheskin/.

[33] Palmatier, R.W., Dant, R.R., Grewal, D., and Evans, K.R. (2006), "Factors Influencing the Effectiveness of Relationship Marketing: A Meta-Analysis", Journal of Marketing, vol. 70 No. 4, pp. 136-53.

[34] Raven, P. and Welsh, D. H.B. (2004), "An Exploratory Study of Influences on Retail Service Quality: A Focus on Kuwait and Lebanon", Journal of Services Marketing, vol. 18 No. 3, pp. 198-214.

[35] Schibrowsky, J.A., Peltier, J.W., and Nill, A. (2007), "The State of Internet Marketing Research - a Review of the Literature and Future Research Directions", European Journal of Marketing, vol. 41 no. 7-8, pp. 722-33.

[36] Scott, L. (1980), "Interpersonal trust: A comparison of attitudinal and situational factors", Human Relations, vol. 33 No. 11, pp. 805-812.

[37] www.secondlife.com

[38] SPSS (1999), SPSS Advanced Models 9.0, SPSS Inc, Chicago, IL.

[39] Surakka, V. and Hietanen, J. (1998), "Facial and emotional reactions to Duchénne and non-Duchénne smiles", International Journal of Psychophysiology, vol. 29, pp. 23-33.

[40] Tsai, W. (2001), "Determinants and consequences of employee displayed positive emotions", Journal of Management, vol. 27 no. 4, pp. 497-510.